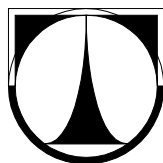


**TECHNICKÁ UNIVERZITA V LIBERCI**  
Fakulta mechatroniky, informatiky a mezioborových studií



## DIPLOMOVÁ PRÁCE

Liberec 2009

**Marián Lamr**

---

# **TECHNICKÁ UNIVERZITA V LIBERCI**

Fakulta mechatroniky, informatiky a mezioborových studií

Studijní program: X2612 – Elektrotechnika a informatika

Studijní obor: 1234T567 – Název studijního oboru

## **Systém využívající mobilního záznamového zařízení obrazu**

## **A system using a mobile image storing device**

### **Diplomová práce**

Autor:	<b>Marián Lamr</b>
Vedoucí práce:	Ing. Přemysl Svoboda
Konzultant:	Ing. Tomáš Martinec

**V Liberci 27. 5. 2009**

## Zadání bakalářské / diplomové práce

<b>Příjmení a jméno studenta</b> (osobní číslo - nepovinné)	Lamr Marián
<b>Zkratka pracoviště</b>	MTI
<b>Datum zadání BP/DP</b>	31. 10. 2008
<b>Plánované datum odevzdání</b>	29. 5. 2009
<b>Rozsah grafických prací</b>	Dle potřeby dokumentace
<b>Rozsah průvodní zprávy</b>	cca 40 – 50 stran
<b>Název BP/DP (česky)</b>	Systém využívající mobilního záznamového zařízení obrazu
<b>Název BP/DP (anglicky)</b>	
<b>Zásady pro vypracování BP/DP</b> (text nijak neformátujte, pouze očísľujte jednotlivé body .. 1) ... 2) ... atd. a každý bod uveďte jako nový odstavec):	
1) seznámte se s problematikou mobilních zařízení, jejich hw a sw vybavení 2) podle uvážení zvolte programovací jazyk a vývojové prostředí 3) navrhnete způsob záznamu obrazu pomocí mobilního zařízení s externí/interní kamerou 4) napište aplikaci, která bude zpracovávat uložené snímky/video	
<b>Seznam odborné literatury</b> (text nijak neformátujte, pouze každou položku uveďte jako nový odstavec):	
Windows Embedded Community, <a href="http://www.windowsfordevices.com">URL:&lt;http://www.windowsfordevices.com&gt;</a> [online][22.10.2008]	
Váš průvodce světem Windows Mobile, <a href="http://smartmania.mobilmania.cz">URL:&lt;http://smartmania.mobilmania.cz&gt;</a> [online][22.10.2008]	
Palmare, <a href="http://palmare.idnes.cz">URL:&lt; http://palmare.idnes.cz/&gt;</a> [online][22.10.2008]	
J. Sharp, J. Jagger, Microsoft Visual C# .NET krok za krokem, Mobile media. Brno 2002.	
Windows Embedded Community, <a href="http://msdn.microsoft.com/en-us/windowsmobile/default.aspx">URL:&lt; http://msdn.microsoft.com/en-us/windowsmobile/default.aspx &gt;</a> [online][22.10.2008]	
<b>Vedoucí BP/DP</b>	Ing. Přemysl Svoboda
<b>Konzultant BP/DP</b> (u externích pracovníků uveďte plný název pracoviště – firmy)	Ing. Tomáš Martinec

## **Prohlášení**

Byl(a) jsem seznámen(a) s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 o právu autorském, zejména § 60 (školní dílo).

Beru na vědomí, že TUL má právo na uzavření licenční smlouvy o užití mé diplomové práce a prohlašuji, že **s o u h l a s í m** s případným užitím mé diplomové práce (prodej, zapůjčení apod.).

Jsem si vědom(a) toho, že užít své diplomové práce či poskytnout licenci k jejímu využití mohu jen se souhlasem TUL, která má právo ode mne požadovat přiměřený příspěvek na úhradu nákladů, vynaložených univerzitou na vytvoření díla (až do jejich skutečné výše).

Diplomovou práci jsem vypracoval(a) samostatně s použitím uvedené literatury a na základě konzultací s vedoucím diplomové práce a konzultantem.

Datum

Podpis

## **Poděkování**

Touto cestou bych chtěl v první řadě poděkovat vedoucímu práce, Ing. Přemyslu Svobodovi za podněty, připomínky, cenné rady a ochotu v průběhu tvorby diplomové práce.

Za finanční podporu v době studia velice děkuji Nadaci paní Olgy Havlové, zvláště vždy ochotné a milé vážené paní Kvasničkové.

V neposlední řadě patří díky za podporu při studiu mé matce, přítelkyni a všem ostatním, kteří mě v době tvorby práce podporovali.

## **Abstrakt**

Tato diplomová práce se zabývá problematikou mobilních zařízení. Jsou zde popsány principy těchto zařízení, jejich hardwarové a softwarové vybavení. V práci je kladen důraz na možnosti týkající se vývoje software pro zařízení s operačním systémem Windows Mobile, komunikace zařízení s databázovým serverem a prezentace získaných multimediálních dat na webových stránkách. Stěžejním bodem práce je návrh systému využívajícího zmíněné principy a technologie. V práci je též uvedena analýza požadavků na systém v podmínkách skutečné realitní kanceláře, vývoj aplikace a implementace celého systému.

## **Klíčová slova**

PDA, Windows Mobile, .NET Compact Framework, C#, GPS, kamera, SQL, Visual Studio, MS SQL Server, ADO.NET

## **Abstract**

This diploma thesis deals with the area of mobile pocket devices, where the principles of these device, their hardware and software equipment are described. The emphasis is put on the possibilities of software during development for the device with Windows Mobile software, communication of the device with the database server and presentation of collected multimedia data on the web sites. The crucial part of this thesis is design of the system using of mentioned principles and technologies. Further is described the analysis of the requirements for the system in the condition of the real estate agency, development of the application and implementation of the whole system.

## **Keywords**

PDA, Windows Mobile, .NET Compact Framework, C#, GPS, Camera, SQL, Visual Studio, MS SQL Server, ADO.NET

# Obsah

1. Úvod.....	10
2. Principy mobilních zařízení .....	11
2.1 Hardware mobilních zařízení .....	13
2.1.1 Základní hardwarové vybavení .....	13
2.1.2 Bezdrátové technologie .....	14
2.1.3 Další užitečný hardware .....	16
2.1.4 Externí kamera.....	18
2.2 Softwarové platformy mobilních zařízení.....	19
2.2.1 Windows Mobile (CE) .....	19
2.2.2 Palm OS .....	23
2.2.3 Symbian .....	24
2.2.4 OS X pro iPhone.....	24
2.2.5 Google Android .....	24
2.2.6 Web OS .....	25
3. Vývoj software pro mobilní zařízení s WM .....	26
3.1 Platforma .NET .....	26
3.1.1 .NET Framework .....	26
3.1.2 .NET Compact Framework .....	29
3.2 Platforma Java ME a jazyk Java .....	30
3.3 Vývojová prostředí a programování pro WM.....	31
3.3.1 Visual Studio 2008 .....	31
3.3.2 Vývoj mobilních aplikací ve VS 2008 .....	32
3.3.3 Windows Mobile 6 SDK .....	33
3.4 Activesync a Windows Mobile Device Center .....	34
3.5 MS SQL server 2008.....	35
3.6 MS SQL Server CE.....	36
3.7 Mobilní aplikace a databáze.....	37
3.8 Komunikace mobilní aplikace s MS SQL.....	38
3.9 Nasazení aplikací pro SQL CE .....	38
3.10 Možnosti synchronizace MS SQL CE s MS SQL .....	38
3.11 Vzdálený přístup k databázi pomocí ADO .NET.....	39
3.11.1 Zprostředkovatelé přístupu k databázi.....	39
3.11.2 Architektura ADO.NET .....	40

3.12 Merge synchronizace .....	42
3.12.1 Konfigurace Publikací .....	43
3.12.2 Konfigurace SQL a IIS Web Serveru pro webovou synchronizaci.....	43
3.12.3 Konfigurace SQL CE .....	43
3.13 Programování webových aplikací .....	44
3.13.1 PHP .....	44
3.13.2 ASP .NET .....	46
3.13.3 Porovnání PHP a ASP .....	46
4. Analýza problému .....	47
4.1 Běžný chod realitní kanceláře .....	47
4.2 Princip systému .....	48
4.3 Analýza požadavků na systém .....	50
4.4 Nutné technické požadavky na systém .....	50
4.4.1 Mobilní zařízení.....	50
4.4.2 OS mobilního zařízení a používaný software.....	52
4.5 Výběr platformy a vývojových prostředků .....	53
4.6 Nezbytné konfigurace .....	53
4.6.1 Konfigurace SQL Serveru .....	53
4.6.2 Konfigurace emulátoru .....	56
5. Vývoj aplikace MobileReality .....	57
5.1 Popis tříd a knihoven.....	57
5.2 Funkce a scénáře mobilní aplikace .....	58
5.2.1 Nový záznam .....	58
5.2.2 Náhled databáze.....	62
5.2.3 Nastavení aplikace.....	63
5.3 Spolupráce s kamerou .....	64
5.4 Ovládání GPS .....	66
5.5 Komunikace s SQL Serverem .....	67
6. Webová aplikace .....	68
6.1 Konfigurace web serveru pro přístup k MS SQL.....	68
6.2 Přehled php skriptů .....	68
6.3 Odesílání požadavků na databázi .....	69
6.4 Optimalizace načítání fotografií z databáze .....	70
7. Závěr .....	71



## Seznam použitých zkratk a termínů

API	-Application Programming Interface
ARM	-Acorn RISC Machine
ASP	-Active Server Pages
CLR	-Common Language Runtime
CLS	-Common Language Specification
CTS	-Common Type System
EDGE	-Enhanced Data for GSM/GPRS Evolution
GPRS	-General Packet Radio Service
GPS	-Global Positioning System
GSM	-Global System for Mobile
GUI	-Graphic User Interface
IIS	- Internet Information Services
IMAP	- Internet Message Access Protocol
IrDa	-odvozeno z Infrared Data Association
JIT	-Just In Time
MDA	-Mobile Digital Assistant
MMS	-Multimedia Messaging Service
MSIL	-Microsoft Intermediate Language
ODBC	- Open DataBase Connectivity
PAL	- Publications Access List
PEAR	- PHP Extension and Application Repository
PECL	- PHP Extension Community Library
PDA	-Personal Digital Assistant
PHP	-Personal Home Page,
PPC	-Pocket Personal Computer
RISC	-Reduced Instruction Set
RAM	-Random Access Memory
ROM	-Read Only Memory
SD, SDHC	-Secure Digital, Secure Digital High Capacity
SDIO	-Secure Digital Input Output
SMS	-Short Message Service
USB	-Universal Serial Bus
UVC	-USB Video Class
WM	-Windows Mobile
XML	- eXtensible Markup Language

## Seznam obrázků

Obr. 1 : Přídavná SDIO zařízení .....	17
Obr. 2 : Časová řada Windows CE .....	21
Obr. 3 : Architektura Windows CE .....	23
Obr. 4 : Platforma .NET .....	28
Obr. 5 : Kompilace programu v prostředí .NET .....	29
Obr. 6 : Prostředí a Smart Device projekty VS 2008.....	33
Obr. 7 : Architektura MS SQL.....	36
Obr. 8 : Struktura DB dat v paměti .....	41
Obr. 9 : Komunikace Mobilní aplikace MS SQL .....	42
Obr. 10 : Princip systému MobileReality .....	49
Obr. 11 : Konfigurace ActyveSync a emulátoru.....	56
Obr. 12 : Úvodní obrazovka .....	58
Obr. 13 : Výběr nemovitosti .....	59
Obr. 14 : Ústřední formulář .....	59
Obr. 15 : CaptureDialog WM 6(emulátor) .....	60
Obr 16 : OpenFileDialog WM 6(emulátor) .....	60
Obr. 17 : Náběrový list (popis) .....	61
Obr. 18 : Náběrový list (Info o majiteli).....	61
Obr. 19 : Online procházení databáze.....	62
Obr. 20 : Nastavení kamery .....	63
Obr. 21 : Nastavení dat .....	63

## Seznam tabulek:

Tabulka 1: Dostupné varianty Palm PC.....	12
Tabulka 2: UVC web kamery .....	18
Tabulka 3 Platformy, procesory a varianty WM podporované VS 2008 .....	32

# 1. Úvod

V dnešním moderním „digitálním světě“ se v mnoha běžných životních situacích stále častěji setkáváme s různými formami počítačů. Stále oblíbenějšími se stávají mobilní kapesní zařízení, která nabývají mnoha podob (PDA, MDA, Smartphone atd.). Zejména v poslední době přicházejí na svět nové obdoby kapesních počítačů a současně s tím se na trhu operačních systémů pro PDA objevují nové ambiciózní projekty. Díky integraci mnoha užitečných technologií a multimediálních funkcí je využití mobilních digitálních asistentů možné jak v oblastech pracovního, tak i osobního života.

Práce si klade za cíl navrhnout způsob záznamu snímků za pomoci mobilního zařízení a vytvořit aplikaci zpracovávající dané snímky. V praxi bychom našli spoustu případů využití takovéto aplikace. Jednou z možností bylo nasazení zmiňované aplikace na realitním trhu, tedy v podmínkách realitní kanceláře, jejíž prostředí a fungování bylo autorovi známo.

Tak vznikla idea vytvořit komplexní systém, který by sloužil pro zjednodušení a větší efektivitu práce makléřů a administrativních pracovníků realitní kanceláře. Více informací o fungování systému se čtenář dozví v kapitole zabývající se principem systému MobileReality.

Celá práce je koncipována do několika kapitol, které spolu úzce souvisí. Hned druhá kapitola je věnována principům mobilních zařízení, jejich hardwarovému a softwarovému vybavení. Ve třetí kapitole jsou popsány možnosti při programování aplikací pro nejrozšířenější platformu Windows Mobile. Čtvrtá kapitola analyzuje daný problém, popisuje princip systému a jsou zde vytyčeny nezbytné požadavky na jednotlivé části systému. Pátá kapitola se zabývá samotným vývojem aplikace MobileReality. Je zde popsána spolupráce s kamerou, GPS modulem, komunikace s SQL serverem a nastíněny jsou scénáře použití mobilní aplikace. V předposlední, šesté kapitole je zmíněna tvorba a implementace webové aplikace prezentující výsledky systému MobileReality. Poslední kapitola je věnována závěru práce.

## 2. Principy mobilních zařízení

Prvně je třeba uvést si definice pro základní parametry kapesního počítače. Hmotnost kapesního počítače se v průměru pohybuje mezi 100 a 250 gramy, rozměry obvykle nepřesahují 15x10x2 cm. Kapesní počítač je díky vlastnímu zdroji napájení vysoce mobilním a kompaktním zařízením. Na základě jednoduchého ovládání pomocí tužky (tzv. stylusu) a dotykového displeje nebrání nic rychlému používání daného přístroje. Tato zařízení nemají klávesnice, které jsou běžné pro stolní PC. To je i důsledek používání mobilního zařízení, které není určeno pro psaní dlouhých zpráv, nýbrž pro zapisování poznámek, adres či krátkých popisků. Důležitou výhodou oproti klasickému stolnímu PC je menší časová prodleva při startu zařízení. Ta je důsledkem toho, že se při každém startu nezapíná operační systém a tím nedochází ke zpomalení startu počítače. Osobní digitální pomocník může komunikovat s okolím jak prostřednictvím propojovacího kabelu, tak pomocí bezdrátových technologií. Základní funkcí je synchronizace, která umožňuje výměnu dat mezi kapesním a stolním počítačem. Lze konstatovat, že kapesní počítač je vlastně počítač, který má procesor, paměť a operační systém, a je pro něj možné programovat různé aplikace.

V dnešní době se velice rychle stírají rozdíly mezi jednotlivými druhy mobilních zařízení. Existují vedle sebe zařízení, která slouží ke stejnému účelu a liší se pouze pojmenováním na základě svého původu. Postupem času a vlivem konkurenčního boje některé druhy přibýly a jiné názvy se naopak vytráčí. Dále budou uvedena nejčastěji používaná názvosloví mobilních zařízení.

### **Palm PC**

Donedávna byl jedničkou na trhu s chytrými mobilními zařízeními. První generace palmů byly vyrobeny v roce 1996. Zprvu Palm používal procesory *DragonBall*, což byly deriváty procesoru Motorola 68000. V pozdější době se ale začaly používat především procesory Intel *Xscale*, které vycházejí z ARM architektury. Postupem času docházelo na základě vývoje k proměně původních Palm PC na Smartphone. Příkladem toho mohou být tzv. *Treo SmartPhones*. Operační systém pro Palm PC byl původně Palm OS. V pozdější době se začal Palm OS nahrazovat systémem Windows Mobile a Palm PC postupně opouštěl přední příčky v oblíbenosti mezi

uživatelé. V tabulce 1 jsou uvedeny poslední modely těchto zařízení, které je možné v současnosti pořídit. Pod pojmem *Handhelds* si lze představit obdobu Pocket PC.

<b>Handhelds</b>	<b>Treo smartphones</b>
Z22	Treo Pro
Zire 72	Treo 800w
Tungsten E2	Treo 500
TX	Treo 500v

**Tabulka 1: Dostupné varianty Palm PC**

## **Pocket PC**

Podle společnosti Microsoft je Pocket PC kapesní zařízení, které umožňuje uživatelům ukládat a načíst emaily, kontakty schůzky, úkoly, přehrávání multimediálních souborů, výměnu textových zpráv, procházení webu a další. Společnost Microsoft stanovuje následující požadavky na Pocket PC:

- Spuštění operačního systému Windows Pocket PC nebo Windows Mobile
- Zařízení má touchscreen nebo touchpad
- Jsou k dispozici hardwarová tlačítka
- Přístroj je založen na ARM architektuře verze 4 a kompatibilní

Pojem Pocket PC prošel jako Palm PC různými fázemi vývoje. V roce 2007 s příchodem nového OS Windows Mobile 6 se Microsoft vzdal tohoto názvu a začal používat značení nová vycházející z verze použitého OS. Přístroje bez integrovaného telefonu nazývá Microsoft Windows Mobile Classic namísto zařízeného Pocket PC. Zařízení s integrovaným telefonem a dotykovou obrazovkou mají název Windows Mobile Professional přístroje a zařízení bez dotykové obrazovky nazývá jako Windows Mobile standard přístroje.

## **PDA**

Zjednodušeně bychom mohli říci, že název PDA (Personal Digital Assistant), česky osobní digitální asistent, v sobě zahrnuje přístroje Palm PC a Pocket PC. Toto označení je používáno v souvislosti s mobilním zařízením bez ohledu na to jaký používá operační systém.

## **MDA/Komunikátor**

Zkratka MDA (Mobile Data Asistent) se překládá jako mobilní datový asistent. Vzniká zde několik otázek týkajících se rozdílů mezi PDA a MDA. MDA je PDA zařízení vybavené GSM modulem, který z něj dělá telefon. V telekomunikačním slangu je slovo komunikátor používáno spíše pro MDA s hardwarovou klávesnicí, i když se s tímto označením můžeme setkat i u mobilních telefonů s klávesnicí. Mezi komunikátory se řadí např. Palm Treo 650, Nokia 9x00, T-Mobile MDA.

## **Smartphone**

Smartphone (chytrý telefon) je zařízení primárně určené pro telefonování, ale s možností si doinstalovat program, přečíst si dokument, stáhnout poštu či podívat se na web. Typickým představitelem Smartphone je třeba Nokia N70, Motorola MPx220 a další. Od MDA se odlišuje především absencí dotykového displeje. Hlavní rozdíl mezi Smartphone a klasickým mobilním telefonem (dále jen MT) je v možnostech jak nahrávat nové aplikace. V případě MT je jediná možnost a to JAVA, která má však určitá omezení. Na Smartphone lze nahrát jakýkoliv software pro např. Pocket PC. Na Smartphone lze pomocí emulátoru spouštět JAVA aplikace, ale naopak to možné již není.

## ***2.1 Hardware mobilních zařízení***

Tato kapitola se zabývá oblastí týkající se hardwaru kapesních počítačů. Budou zde uvedeny informace o základním a dalším hardwaru rozšiřujícím možnosti kapesních počítačů.

### **2.1.1 Základní hardwarové vybavení**

#### **Procesory**

Většina PDA používá procesory typu RISC (Reduced Instruction Set Computer). Do této kategorie spadají procesory typu MIPS či ARM, jejichž mutace lze v PDA zařízeních najít. V posledních letech se stala nejoblíbenější architekturou v této oblasti ARM a téměř vytlačila MIPS architekturu z tohoto segmentu. Důvodem jejich použití je zejména skutečně nízké vyzařované teplo, kdy si procesory naprosto vystačí s pasivním

chlazením bez přehřívání zařízení. Zároveň mají nízkou spotřebu, a to vše při dostatečném výpočetním výkonu. Frekvence těchto procesorů se v dnešní době pohybuje od 200 až po 1000 MHz.

### **Paměti a datová úložiště**

Kapesní počítače mají dva druhy paměti a to RAM a Flash ROM. Paměť RAM (Random Acces Memory) slouží jak pro potřeby operačního systému, tak i pro běh samotných aplikací. Samozřejmě, že data uložená v RAM jsou po vypnutí zařízení ztracena, tzn., že RAM je závislá na přísunu elektrické energie. Velikost RAM dnešních kapesních počítačů obvykle začíná na 64 MB. Tato velikost je opravdu základním minimem, ale pro běh cca 4 nenáročných aplikací je vyhovující. Z těchto 64 MB paměti zůstává uživateli k dispozici přibližně 25 – 35 MB. Na trhu jsou však dostupná zařízení, která disponují pamětí RAM i 512 MB. Záleží na potřebách uživatele, jaké záměry s daným kapesním počítačem má.

Paměť Flash ROM (Read Oly Memory) slouží k uložení operačního systému a případně programů a uživatelských dat. Na rozdíl od paměti RAM nepotřebuje tento typ paměti stálý přísun energie, tzn., že při vyjmutí akumulátoru uživatel nepřijde o svá data. Kapacity ROM se pohybují mezi 64 – 512 MB.

Jako úložiště většího množství dat se pro PDA používají nejčastěji CF, Micro SD či Micro SDHC paměťové karty, které lze použít pro umístění programů i uživatelských dat. Výhodou paměťových karet je snadná přenositelnost a možnost použití jako externí úložiště dat, jelikož se dají snadno do PDA připojit pomocí microSD slotu. V případě využití paměťových karet větších než 4 GB, musí PDA podporovat technologii SDHC. Dnes se běžně můžeme setkat i s kartami o velikosti 16 GB.

## **2.1.2 Bezdrátové technologie**

### **Infračervený port**

IrDA periferie umožňuje zařízením komunikovat mezi sebou díky LED diodě, která vysílá infračervené záření. Existují dva standardy IrDA a to verze 1.0, která disponuje přenosovou rychlostí 115 kb/s a verze 1.1 která má přenosovou rychlost 4

Mb/s. Při přenosu dat je nutná přímá viditelnost obou zařízení, což je velice nepohodlné. V současnosti je tato technologie dosti na ústupu.

## **Bluetooth**

Bluetooth je populární bezdrátová technologie, která se vyskytuje ve většině moderních PDA, Smartphone a mobilních telefonech. Pomocí této radiové technologie pracující na frekvenci 2,4 GHz můžeme vytvořit osobní počítačovou síť. Dnes se setkáváme v kapesních počítačích s verzí 2.0 či 2.1 s rozšířením EDR (Enhanced Data Rate) jež umožňuje dosáhnout přenosové rychlosti až 2,2 resp. 3 Mbit/s. K 21. dubnu 2009 byla přijata specifikace 3.0, která bude umožňovat rychlost přenosu až 24 Mbit/s. Výkonnost bluetooth lze rozdělit do 3 kategorií a to: Class 1 s maximálním dosahem 100m, Class 2 s dosahem 10m a Class 3 s dosahem 1 m.

## **Wifi**

Wifi(Wireless fidelity) vychází ze specifikace IEEE 802.11 a slouží k zajištění vzájemného bezdrátového propojení přenosných zařízení a jejich připojování na lokální síť LAN. Postupem času se Wifi stává standardní výbavou nových modelů PDA. Chceme-li připojit PDA do sítě pomocí Wifi, musí dané zařízení obsahovat integrovaný či externí Wifi-adaptér a poté je nutné projít průvodcem konfigurace bezdrátové sítě. Nemá-li PDA interní adaptér, lze externí adaptér připojit pomocí slotu SD karty. Ten však slouží pouze k napájení a samotná komunikace mezi Wifi modulem a PDA probíhá pomocí Bluetooth.

## **GSM/GPRS/EDGE modul**

GSM lze překládat jako systém pro mobilní komunikaci. Na základě modulu GSM/GPRS mohou kapesní počítače plnit funkci mobilního telefonu. Jak již bylo řečeno, PDA s daným modulem je nazýváno MDA (Mobile Data Assistant). Díky tomuto modulu lze přistupovat ke službám, jakými jsou např. klasické telefonní hovory, SMS zprávy, multimediální zprávy MMS, ale také přístup k internetu pomocí technologie GPRS či EDGE. Vybraná přenosná zařízení disponují službami mobilních sítí třetí generace (3G)



## **GPS moduly**

GPS modul dnes ještě nebývá standardní součástí PDA, i když se objevuje stále častěji. Bezdrátová technologie GPS, neboli Global Position System, slouží pro určení polohy na zemském povrchu. Součástí systému jsou samotné družice (celkem 24), řídicí střediska a uživatelská část. Pod uživatelskou částí je třeba si představit samotné uživatele, kteří se pohybují na Zemi. Určení polohy probíhá na základě příjmu signálu alespoň od 3 družic. Chceme-li zjistit i nadmořskou výšku, je zapotřebí příjmu od 4 družic. K samotnému výpočtu dochází pomocí tzv. pseudo-vzdálenosti, což je vzdálenost přijímače od družice a znalost zpoždění, za jakou dobu signál z družice dorazí do přijímače. Opět jsou dvě možnosti, jak získat přístup k GPS technologii v PDA. Je možné mít interní zabudovaný GPS modul, či lze k připojení externího GPS modulu použít Bluetooth či USB kabel.

### **2.1.3 Další užitečný hardware**

Mezi další užitečný hardware, který rozšiřuje multimediální možnosti PDA, ulehčuje psaní textu či rozšiřuje konektivitu a možnosti použití dalších přídatných zařízení bychom měli jmenovat Interní kameru, hardwarové klávesnice, USB host a SDIO slot.

#### **Interní kamera**

Interní kamera je obsažena téměř v každém chytrém, moderním mobilním zařízení. Kvalita videa a fotografií je různá. Nejběžnější rozlišení fotoaparátů se pohybuje v okolí 1 či 2 Mpx. Je možné v kapesních počítačích najít i kvalitnější fotoaparáty. Za ty si ale musíme připlatit.

#### **SDIO slot**

Slot pro paměťové SD karty dnes velmi často podporuje technologii SDIO(Secure Digital Input Output). V takovém případě je možné kromě paměťových karet připojit k PDA libovolné další přídatné zařízení vybavené SDIO kartou. Operační systém velmi často nativně podporuje tyto přídatné karty. Z těchto rozšiřujících zařízení jmenujme alespoň Wifi, bluetooth, GPS moduly, TV tuner, či čtečku otisků prstů. Názorné příklady takovýchto zařízení jsou vyobrazeny na obrázku 1.



**Obr. 1 : Přídavná SDIO zařízení [1]**

## **USB host**

Velice šikovným řešením je USB host hardwarová periferie, pomocí níž je možné k PDA připojit velké množství různých zařízení. Standardní USB jak ho známe z PC, pracuje na principu master-slave. USB host u PDA pracuje jako master a připojené zařízení se chová jako slave. USB host v PDA inicializuje veškeré dění a přenos dat, kdežto zařízení připojené, může pouze přijímat a provádět instrukce. Existují zařízení, která připojíme k PDA bez problémů jako například USB Flash disk či fotoaparát pro prohlížení, zálohu a kopírování souborů. Dále je možné připojit k USB host periférii USB hub a tím máme možnost komunikovat s několika zařízeními najednou. Tato možnost je ovšem PDA velice energeticky náročná. Jiná zařízení specifická svým použitím potřebují k ovládání PDA přístrojem speciální ovladače pro nainstalovaný OS. Příkladem toho může být i připojení externí kamery, která ovšem musí splňovat další požadavky, o kterých bude pojednáno později.

#### 2.1.4 Externí kamera

Jelikož je tato problematika součástí tohoto projektu, je pro ni vyhrazena samostatná kapitola a bude jí věnována větší pozornost i v dalších kapitolách. V případě, že bychom chtěli připojit k PDA externí kameru je situace poněkud složitější. Tato kapitola se pokusí nastínit pouze základní požadavky na externí kameru bez ohledu na softwarovou stránku problému. V první řadě je třeba si rozmyslet, jaký typ externí kamery k PDA je možné vůbec připojit a co se od daného řešení očekává. Již z principu je jasné, že tak malé zařízení jako je kapesní počítač nebude schopné energeticky uspokojit externí kameru na delší dobu. To znamená, že připojení kamery by mělo být krátkodobé, k rychlému pořízení snímku a kamera by měla jít jednoduše odpojit. Dalším problémem je periferie, přes kterou by se mohla připojit externí kamera. Jediným prakticky možným řešením zůstává USB host periferie, který umožňuje pohodlné připojení a odpojení kamery, řídí veškeré dění a zasílá požadavky kameře. Současně bude také USB představovat i napájení pro kameru. Velice důležitý požadavek na USB externí kameru je, aby kamera patřila do tzv. UVC (USB Video Class). Tyto požadavky na externí kameru splňuje celá řada webových kamer pro stolní PC, avšak není pravidlem, že nové modely webkamer jsou standardu UVC. Následující tabulka obsahuje dnes běžně dostupné UVC webkamery na českém trhu.

UVC Web kamery běžně dostupné na českém trhu	
Creative Live! Cam Video IM Ultra	Logitech QuickCam Pro 9000
Creative Live! Cam Optia AF	Logitech QuickCam for Notebooks Pro
Creative Live! Cam Optia Pro	Logitech QuickCam S5500 for Business
Genius VideoCam iSlim 320	Microsoft LifeCam NX-6000
Genius VideoCam iSlim 321R	Philips SPC 2050NC
Genius VideoCam iSlim 1320	Philips SPC 530NC
Genius VideoCam eYE 320SE	Philips SPC 620NC
Labtec WebCam 5500	Philips SPC 630NC
Labtec WebCam 3300	Samsung Pleomax Acryl W-400W
Labtec WebCam 2200	Trust WB-6250X Webcam
Logitech QuickCam 3000 for Business	Trust USB2 Auto Focus Webcam
Logitech Quickcam E 3500	

**Tabulka 2: UVC web kamery**

Tento průzkum byl tvořen k datu 5. 5. 2009 v 13-ti největších E-shopech s PC komponentami a elektronikou v ČR. Při hledání byla zkoumána specifikace výrobce daného produktu.

V případě zájmu zjištění UVC kompatibility o další modely a značky, které nesou běžně k dostání na českém trhu lze shlédnout následující webové zdroje.

[http://www.quickcamteam.net/devices/logitech\\_uvc\\_device\\_list.pdf](http://www.quickcamteam.net/devices/logitech_uvc_device_list.pdf)

[http://en.wikipedia.org/wiki/List\\_of\\_USB\\_video\\_class\\_devices](http://en.wikipedia.org/wiki/List_of_USB_video_class_devices)

Operační systémy pro mobilní zařízení standardně nepodporují UVC zařízení a je potřeba k jejich ovládání speciální ovladač. O dostupnosti těchto ovladačů a případné vlastní svépomoci bude řečeno v kapitole 4.4.1

## ***2.2 Softwarové platformy mobilních zařízení***

V této kapitole bude pojednáno o pěti nejpopulárnějších operačních systémech, které se používají v současné době na kapesních počítačích a Smartphonech. Budou zde popsány základní principy a rozdíly mezi jednotlivými operačními systémy a směry, kterými se dané systémy ubírají. Největší pozornost bude věnována OS Windows Mobile, který je použit jako platforma pro mobilní zařízení tohoto diplomového projektu.

### **2.2.1 Windows Mobile (CE)**

Nejprve bude pojednáno o operačním systému Windows Mobile společnosti Microsoft. Tento OS je na mobilních zařízeních typu PDA či MDA dnes nejvíce rozšířen a jeho využití najdeme jak pro osobní potřebu, tak pro pracovní záležitosti či dokonce pro zábavu a multimédia.

#### **Historie a verze Windows mobile**

Historie Windows CE, jak jsou někdy Windows Mobile označovány, sahá do roku 1992, kdy významné společnosti jako Microsoft, Compaq, Motorola NEC a Sharp chtěli implementovat tehdejší Windows z klasických počítačů do počítačů kapesních. Za tímto účelem vznikl projekt s názvem *WinPAD*, který se ovšem díky tehdy ještě

nevyspělým technologiím nepodařilo uskutečnit. V tuto dobu ještě neexistovaly mobilní procesory a tak na vině neúspěchu byly především tehdejší klasické procesory s velikou spotřebou, na které nestačily příliš energeticky slabé baterie. Počátkem roku 1995 byly započaty testy na novém projektu *Pegasus*, do kterého byli zapojeni i odborníci z neúspěšného *WinPAD*. Tento nový projekt používal zcela nová mobilní zařízení, na která byla stanovena dosti přísná specifikace. Měla být použita 32-bit. architektura procesoru založená na MIPS 3000 nebo MIPS 4000, minimálně 4 MB paměti ROM a 2 MB paměti RAM, dotykový displej s rozlišením 480x240, stylus na jeho ovládání, QWERTY klávesnice, Infra port, slot pro PCMCIA karty, sériový port RS-232, výstup pro audio, rozměry 18x10x2,5 cm, napájení na dvě AA baterie, a váha menší než 500g. Operační systém byl nazván Windows CE 1.0.

Dále vyvinutý Windows CE 2 byl první vestavěný OS, tzn., přestal být pouze OS pro kapesní počítače. Výrobci hardware mohli vyrábět zařízení pracující na této platformě, takže se s touto platformou lze setkat i v autech, herních konzolách, automatech atd. V této verzi se již objevily aplikace typu Word, Excel, Outlook, PowerPoint, Internet Explorer. Tento systém již podporoval kromě jiného systém souborů FAT 32, Wifi a USB port.

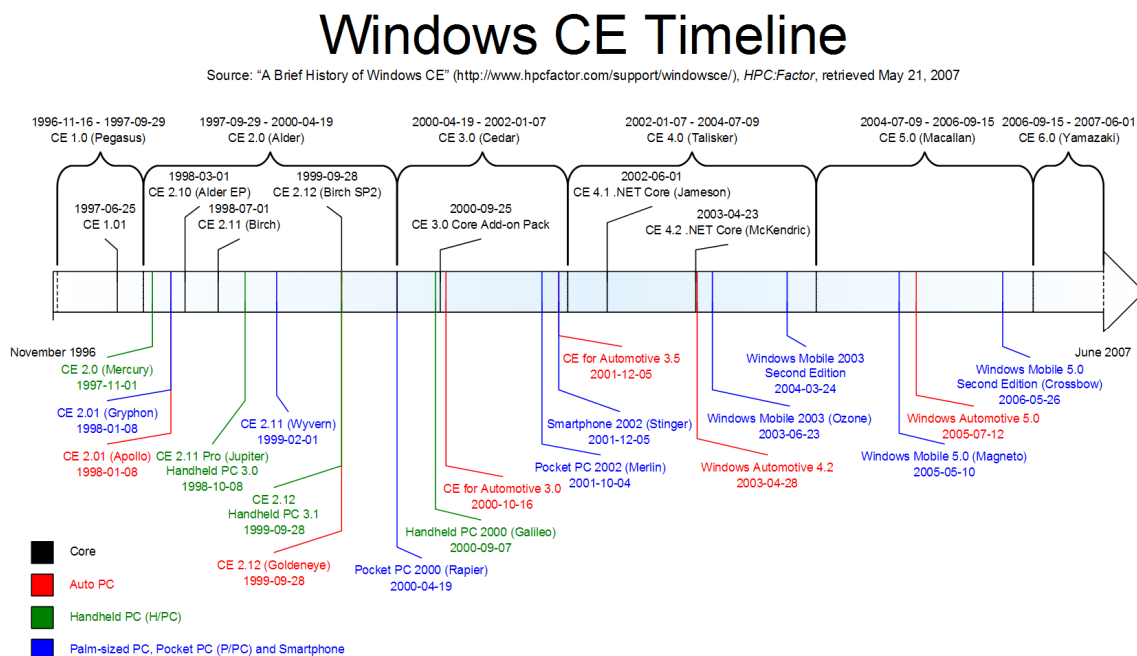
Velkou změnu znamenal příchod Windows CE 3.0, která spatřila světlo světa v prvním čtvrtletí roku 2000 a znamenala příchod prvních PDA. Tato nová PDA byla dosti podobná zařízením Palm, která měla v onu dobu velkou popularitu. Po roce používání byl tento OS pojmenován Pocket PC 2002, a v následujícím roce byla vydána jeho verze i pro Smartphone.

Windows CE 4 jsou opět známé spíše pod označením Pocket PC 2003 a Windows Mobile 2003 SE. Vylepšení v této verzi spočívalo v podpoře procesorů Intel postavených na ARM architektuře. WM 2003 SE se lišila od Pocket PC 2003 podporou rotace obrazovky.

S rokem 2005 přichází na trh nová verze Windows CE 5 (Windows Mobile 2005). Microsoft zde poprvé zpřístupnil část zdrojového kódu vývojářům. Změny oproti minulé verzi byly následující. OS je uložen ve FLASH paměti a načítán je do RAM, OS obsahuje plnou verzi DirectX8, nové Office Mobile, Windows Media Player 10 Mobile, podporu USB 2.0 a GPS.

V roce 2007 byly představeny tři nové verze Windows Mobile 6 (Windows CE 6). Windows Mobile 2006 Standard je určen pro Smartphone. Windows Mobile 2006 Classic najdeme na klasických PDA bez GSM modulu a verze Professional je určena pro MDA,

tedy PDA s telefonním modulem. S těmito verzemi se stávají samozřejmostí funkce jako automatická aktualizace, šifrování dat na paměťových kartách a Windows Live. Samotný systém je oproti starším předchůdcům celkově rychlejší. Vývoj Windows Mobile je graficky znázorněn na Obr. 2.2.



**Obr. 2 : Časová řada Windows CE [2]**

### Aktuální verze a budoucnost Windows mobile

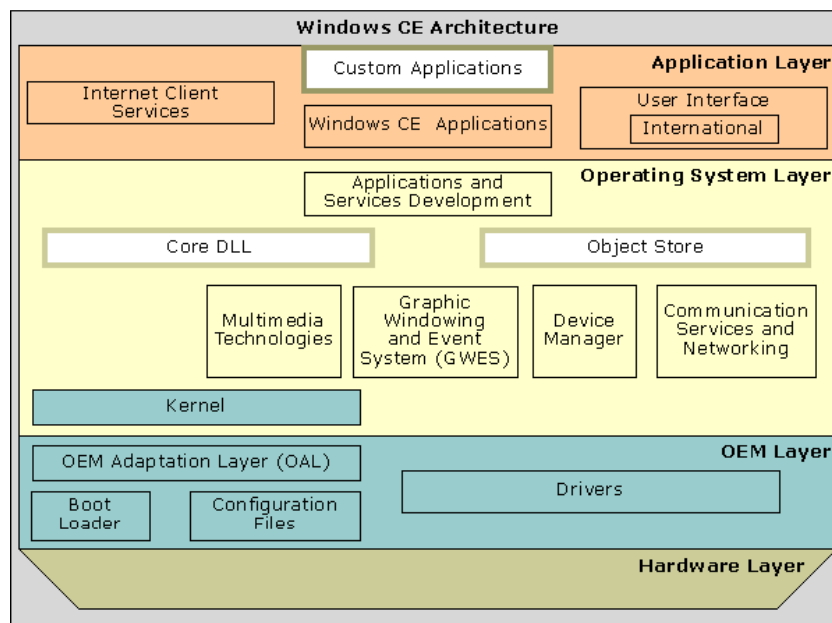
Nadcházející verze Windows Mobile 6.5 se stala během psaní této práce verzí téměř aktuální. Představení finální verze Windows Mobile 6.5 bylo naplánováno na 11. května 2009 na konferenci Microsoft TechEd 2009. V této době by měla být předána výrobcům zařízení, aby mohli připravovat finální verze ROM pro stávající zařízení na trhu, či aby mohli vytvářet ROM s novým systémem WM. Bude záležet pouze na konkrétním výrobcu, zdali zakoupí licence pro svoje produkty, aby mohly stávající zařízení aktualizovat na verzi 6.5. Díky novým funkcím má WM 6.5 větší nároky na paměť RAM a dále bude zmíněný upgrade možný pouze u zařízení s velikostí RAM minimálně 128 MB.

Novinek je ve verzi 6.5 skutečně dost. Jmenujme ty nejdůležitější. Podpora technologií *Silverlight* a *Flash* a zcela nový IE by měly sloužit k pohodlnějšímu

prohlížení internetu. Ovládání GUI pomocí šestiúhelníku neboli tzv. plástve a celkově snadnější ovládání prsty je stejně jako *Market Place* pro stahování aplikací inspirováno úspěchem těchto funkcí u konkurenčního iPhone. Dále jmenujme novou funkci *Myphone* k bezdrátové synchronizaci textových zpráv, kontaktů, fotek a videí. Poslední důležitou novinkou je podpora kapacitních displejů a *multitouch*. Kvůli zjednodušení vnímání různých systémů mezi uživateli se Microsoft rozhodl navenek Windows Mobile přejmenovat na Windows for Phones.

### **Architektura Windows Mobile**

Třicetidvou-bitový operační systém Windows Mobile je otevřený OS s multitaskingem a multithreadingem. Vývojáři mohou vytvářet software pro tuto platformu díky rozhraní pro programování aplikací Win32 API (Application Programming Interface). Pro potřeby implementace na různá hardwarová zařízení se WM skládá z jednotlivých modulů, které mohou být sestavovány nezávislými vývojáři. Základní čtyři vrstvy systému zabezpečují určité části služeb. První, Aplikační vrstva (Application Layer) komunikuje s aplikacemi a jsou v ní definovány služby www, email apod. Vrstva operačního systému (Operating System Layer) se skládá z jádra, objektového skladiště, DLL knihoven, síťových a komunikačních protokolů a Grafického okenního událostního subsystému GWES (Graphic Windowing and Events Subsystem). GWES je vlastně jakýsi interface mezi aplikacemi, uživatelem a OS. Třetí hlavní vrstvou je tzv. OEM (Original Equipment Manufacturer) vrstva. Její součástí je tzv. OAL (OEM Adaptation Layer), která od sebe odděluje jádro a hardwarovou vrstvu, řídí přerušení, implementuje periferie, systémový čas a napájení. Poslední vrstvou je hardwarová vrstva, reprezentující fyzický hardware. Detailní zobrazení jednotlivých vrstev je znázorněno na obrázku 2.3.



**Obr. 3 : Architektura Windows CE [3]**

### 2.2.2 Palm OS

Palm OS je grafický operační systém Společnosti Palm určený pro PDA a komunikátory. Jeho historie sahá do roku 1996 a hlavním zaměřením byly obchodní aplikace. Palm OS prošel postupným vývojem od použití na prvních přístrojích postavených na procesorech Dragonball, až po Architekturu ARM. Palm OS nerozlišoval paměť pro systém souborů a paměť RAM. Všechny soubory byly instalovány přímo do paměti RAM a též z ní i spouštěny. Kapacita této paměti byla od 0,5-1 MB. Tato paměť nebyla dále rozšiřitelná. Od počátku systém podporoval synchronizaci dat přes sériový port s osobním počítačem či jiným zařízením. Postupem času nabízel Palm i podporu paměťových karet SD či MMC. Připojení bylo možné pomocí IrDa, Bluetooth a Wifi. Hlavní výhodou Palm OS byly jeho nízké nároky na výkon a paměť použitého zařízení. Jeho další výhodou byla i jeho rychlost. Na Palm OS mohla běžet od prvních verzí až do verze 5 pouze jedna aplikace. To znamená, že nepodporuje multitasking. Přepínání mezi aplikacemi probíhalo tak, že spuštěná aplikace si při přepnutí na jinou uloží svůj poslední stav, čímž uživatel nic nepozná. Při opětovném přepnutí do původní aplikace se všechna data načtou zpátky. Od verze 5 lze na Palm OS spustit jednu až dvě aplikace jako tzv. rezidentní. Tento OS byl v lednu 2009 nahrazen novým systémem webOS.



### 2.2.3 Symbian

Operační systém Symbian je systém navržený zejména pro mobilní zařízení typu Smartphone, které jsou více telefony než komunikátory. Je potomkem systému EPOC32 který se používal na zařízeních společnosti Psion. Dnes je používán výhradně na procesorech architektury ARM. Existuje několik odnoží Symbianu a ty se od sebe liší tím, pro jaké účely a zařízení jsou určeny. Řada S60 je určena pro telefony klasické konstrukce vyznačující se malou cenou, hmotností a velikostí a jsou snadno rozšiřitelné prostřednictvím dalších aplikací. Hlavní nevýhodou je nekompatibilita s aplikacemi ze starších verzí systému. Řada S80 je uzpůsobena pro pracovní účely a nachází se více u zařízení s klávesnicí, větší konstrukce a hmotnosti, nazývaných komunikátory (Smartphone s klávesnicí). Za zmínku by ještě stála nadstavba UIQ (User Interface Quartz), která podporuje ovládání telefonu přes dotykovou obrazovku. Ta je tedy určena především pro telefony s dotykovou obrazovkou.

### 2.2.4 OS X pro iPhone

OS X pro iPhone je upravená verze operačního systému Mac OS X určeného pro počítače Macintosh. Tento operační systém, který nepodporuje multitasking, využívají zatím pouze dvě mobilní zařízení od výše jmenované firmy a to mobilní telefon Apple iPhone a multimediální přehrávač iPod Touch. Co se samotného iPhone týče, tak lze říci, že je to multimediální mobilní zařízení s kapacitním displejem, GSM modulem a pohybovým senzorem, které umožňuje práci s internetem a multimédií. Výhodou OS X pro iPhone, je podpora ovládání zařízení pomocí dotykové obrazovky bez pomoci stylusu a podpora *multitouch* a gest. Přístroj má velikost paměti jednoznačně danou a nelze ji rozšířit, což je nevýhoda. Také ho standardně nelze připojit k PC jako USB mass storage (zařízení lze používat podobným způsobem jako disketu). Je možné říci, že iPhone je svým použitím určen spíše pro zábavu než pro práci.

### 2.2.5 Google Android

Google Android je open source softwarová platforma založená na Linuxu, která je určena pro mobilní zařízení jako PDA, MDA či Smartphone. Původně tento multitaskingový operační systém vyvinula firma Google, ta jej ale po představení

v listopadu 2007 předala spolu se zdrojovými kódy nově založenému konsorciu Open Handset Alliance, které je členem. Vývojáři mohou psát aplikace pro Android díky SDK Android. Použitým jazykem je JAVA s využitím knihoven vyvinutých Google. SDK pro vývoj obsahuje vývojové prostředí, emulátor, debugger, knihovny a dokumentaci s příklady. Google Android je často srovnáván s o něco starším konkurenčním OS X pro iPhone či novým Web OS společnosti Palm.

### **2.2.6 Web OS**

Nejmladším, ale zato velice ambiciózním hráčem na poli operačních systémů pro mobilní zařízení typu PDA, MDA a Smartphone je právě Web OS. Web OS je multitaskingový operační systém založený na Linuxu a webových technologiích Html, CSS, JAVA atd. Stal se nástupcem operačního systému Palm společnosti Palm inc. Poprvé byl představen v lednu 2009 na výstavě CES. První přístroj nazývaný Palm Pre by měl být dostupný do konce první poloviny roku 2009. Aplikace pro tento OS bude možné vytvářet pomocí Mojo SDK, které bylo uvolněno začátkem května 2009. Díky vestavěnému emulátoru Pro Palm OS by měl mít Web OS podporu spuštění aplikací původně určených pro Palm OS. Web OS směřuje do stejného segmentu jako konkurenční iPhone a dává si za cíl mu konkurovat.

### 3. Vývoj software pro mobilní zařízení s WM

Vývoj aplikací pro zařízení s Windows Mobile je možný buď pomocí platformy .NET nabízené společností Microsoft, ale také je možné vytvářet Java aplikace, které se poté spouští na WM zařízení v emulátoru Javy.

#### 3.1 Platforma .NET

Platforma .NET (nebo také dot NET pochází ze slova network) je soubor technologií, které jsou dostupné nejen pro tvorbu webových aplikací. Její základní komponentou je Microsoft .NET Framework, což je vlastně prostředí, které slouží jako spouštěcí rozhraní. Obsahuje potřebné knihovny a slouží jako prostředí potřebné pro běh aplikací. Cílem této platformy je zpřehlednit zdrojové kódy aplikací a zjednodušit jejich tvorbu. Tato vcelku nová technologie se snaží vzít to nejlepší z ostatních již existujících. Dot NET pracuje s celou řadou programovacích jazyků jako je C# Visual Basic .NET, Delphi, Managed C++,F#, J# či IronPython. Platforma .NET tedy nenařizuje použití žádného konkrétního programovacího jazyka a měla by umožnit programování nejen pro Windows ale i pro OS X a Linux. Pro programování webových a mobilních aplikací jsou nejvýhodnější C# a Visual Basic .NET, které dávají co se týče výkonnosti a funkčnosti stejný výsledek. Moderní a elegantnější je ale C# jehož nevýhoda oproti Visual Basic .NET je jen v obtížněji čitelném kódu.

##### 3.1.1 .NET Framework

Prostředí .NET Framework, se skládá ze tří základních částí a to CLR, CTS a CLS. CLR(Common Language Runtime) je jakési prostředí pro .NET aplikace, které se v něm spouští.

Hlavní rolí CLR je načítání, vyhledávání a správa typů platformy .NET a v neposlední řadě se také stará o správu paměti a provádění bezpečnostních kontrol. Kódu spuštěnému v rámci CLR se říká tzv. managed code, což se do češtiny překládá jako řízený kód. To znamená, že aplikace běžící v CLR se musí řídit specifikacemi určenými pro .NET. Tyto specifikace jsou vlastně dvě zbývající základní části .NET Frameworku CTS a CLS.

CTS (Common Type System) zcela popisuje veškeré možné datové typy a programovací konstrukce, které runtime podporuje. Dále specifikuje, jak mohou tyto entity spolu komunikovat a jak se mají reprezentovat ve formátu metadat.

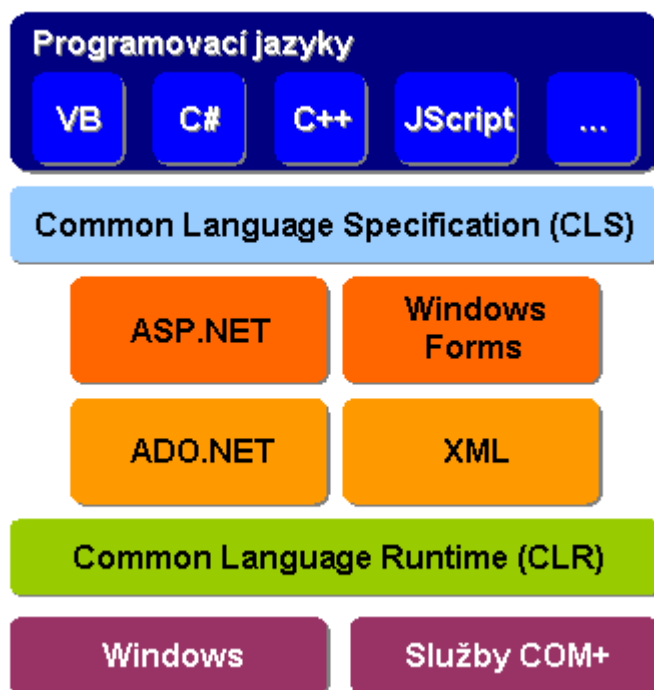
CLS (Common Language Specification) je společná specifikace jazyků, určující povahu vývojového jazyka. Definuje tedy podmnožinu společných typů a programovacích konstrukcí, na kterých se mohou všechny programovací jazyky .NET dohodnout. Prakticky to znamená, že vytváříme-li typy vystavující pouze ty schopnosti, které jsou v souladu s CLS můžeme si být jisti, že je budou moci využívat i všechny jazyky používané v .NET. Takto vytvářené aplikace plně spravuje CLR a ručí za jejich správný běh. Krom spuštění managed kódu je ale možné spustit i takzvaný unmanaged kód (neřízený kód). Neřízený kód se nemusí řídit těmito specifikacemi a jako jeho příklad si můžeme uvést COM komponenty

Shrňme si tedy základní činnosti nejdůležitější části CLR:

- Správa kódu, který v něm běží, správa paměti, „garbage collector“, zachytávání výjimek a kontrola správnosti typů.
- Přístup k systémovým službám Windows API a přístup ke COM službám.
- Propojení aplikací, které byly vyvinuty v různých programovacích jazycích.

Další informace se týkají garbage collectoru. Inspirací při tvorbě specifikace .NET si Microsoft vzal z jazyka JAVA, který má rovněž garbage collector přímo ve své specifikaci. Garbage collector má za úkol automaticky určit, která část paměti programu je již nepoužívána a tuto část paměti automaticky uvolní, poté co na ni zaniknou veškeré reference. Díky garbage collector se není možné odkazovat na položku, která v paměti není a nemůže tak dojít k porušení ochrany paměti. Garbage collector má také své nevýhody. První je větší zatížení procesoru, jelikož garbage collector běží ve dvou vláknech a druhou dosti významnou nevýhodou je, že za normální situace není možné říci, kdy bude daná část paměti odstraněna

Schéma znázorňující platformu .NET a spolupráci jednotlivých částí je vyobrazeno na Obr 2.4.



**Obr. 4 : Platforma .NET [4]**

Nad Common Language Runtime stojí na jedné straně knihovna ADO. NET s ASP .NET, které slouží pro práci s daty, vzdálený přístup a vývoj uživatelského rozhraní pro webové aplikace. Na druhé straně je knihovna Windows Forms společně s XML pro vývoj desktopových a mobilních aplikací.

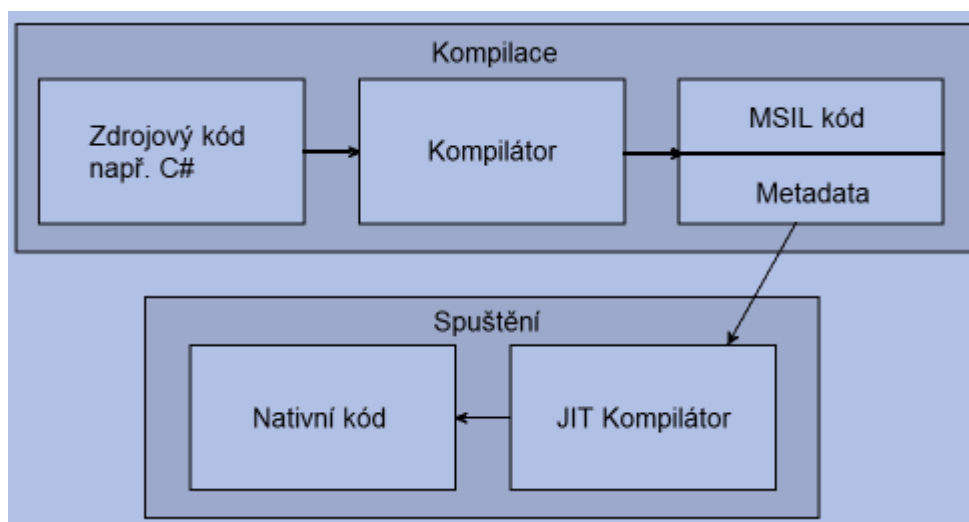
Programy vytvořené na libovolném programovacím jazyku spadajícím pod .NET se překládají do tzv. MSIL kódu. MSIL(Microsoft Intermediate Language) je jazyk podobný assembleru, který byl vyvinut Microsoftem, aby mohli tvůrci programovacích jazyků vytvářet jazyky nové pro platformu .NET. Při kompilaci aplikace vyvíjené v .NET bere překladač jako vstup zdrojový kód daného jazyka a výstupem je program v jazyce MSIL. CLR zajišťuje překlad z jazyka MSIL do strojového kódu. Toto se děje při prvním spuštění aplikace. Překlad probíhá pomocí JIT(Just-in-time) kompilátoru. Samozřejmě, že by bylo možné vyvíjet aplikace i v kódu MSIL, ale to by bylo velice nevýhodné. Existují tři druhy JIT kompilace.

První jmenujme kompilaci v době instalace, což však není pravá JIT kompilace neboť se neprovádí při spuštění aplikace, ale už při jeho instalaci což je výhodné neboť se MSIL kód překládá pouze jednou a při dalších spuštěních se spouští už jen strojový

kód. Nevýhodou této kompilace je, že v případě zkopírování aplikace na jiný počítač ji nelze spustit a musí se nainstalovat znova.

Při skutečném JIT překladač je program spuštěn a MSIL kód je přeložen najednou. Výhodou této kompilace je, že strojový kód je optimalizovaný na konkrétní stroj. Nevýhodou je režie tohoto JIT, neboť se provádí při spouštění aplikace. Po kompilaci je však kód stejně rychlý jako v předchozím případě.

Ekonomický JIT překlad je skutečný JIT překlad. Ovšem při překladač MSIL do strojového kódu jsou vypnuty optimalizace pro zrychlení a zmenšení programu. Tento překlad je vhodný pro zařízení s omezenými zdroji a malou pamětí. Překlad se totiž neprovádí celý, ale jen ta část programu, která je prováděna. Funkce, které nejsou zatím přeloženy, se nahradí krátkým kódem tzv. stubem, který se tváří jako funkce plnohodnotná a až při jeho zavolání se skutečná funkce přeloží.



**Obr. 5 : Kompilace programu v prostředí .NET [6]**

### 3.1.2 .NET Compact Framework

Compact Framework, někdy označován zkratkou CF, je část prostředí .NET Framework. Již z názvu je zřejmé, že Compact Framework je „ořezanou“ verzí .NET Frameworku. Toto prostředí je určeno pro mobilní zařízení typu PDA/MDA, Smartphone či Embedded (vestavěná) zařízení používající některou z mutací operačního systému Windows CE či Windows Mobile. Při vývoji aplikací pro WM přístroje existují dvě možnosti. Buď je možné využít Compact Framework a nebo použít ASP.NET

Mobile Control. ASP.NET Mobile Control je ne příliš používané řešení vzhledem k tomu, že používá architekturu klient-server, což vyžaduje nutně neustálé připojení k síti a není možné používat periferní zařízení. Naopak díky tomu, že Compact Framework neobsahuje odlišné jmenné prostory od .NET Framework, lze v něm tvořit stejně plnohodnotné mobilní aplikace. Vývojové prostředí je integrováno do Microsoft Visual Studio a běhové prostředí (runtime) Compact Framework je do staršího mobilního zařízení třeba doinstalovat. Velikost runtime je přibližně 2 MB. V novějších mobilních zařízeních je Compact Framework runtime (CLR) již nainstalován. Hlavní vlastnosti CLR jako společný typový systém, garbage collector, typově bezpečný runtime, ošetření chyb pomocí výjimek a just-in time kompilace jsou stejné jako u plnohodnotného .NET Frameworku. Asi největším rozdílem mezi .NET Framework a CF je vypuštění mnoha knihoven. CF má přibližně jednu čtvrtinu z celkového počtu knihoven plnohodnotné verze .NET Framework. V CF chybí ještě podpora pro ASP.NET, podpora pro psaní COM/ActiveX objektů, binární serializace a kompilování aplikace v době instalace. Z programovacích jazyků můžeme volit mezi C# a Visual Basic .NET. V Compact Frameworku tyto jazyky nelze v jednom programu kombinovat.

### ***3.2 Platforma Java ME a jazyk Java***

Pro mobilní zařízení je možné vyvíjet software také na platformě Java ME. Java ME (Micro Edition), dříve označována Java2ME, je jedna ze tří platforem Javy. Tato platforma je určena právě pro zařízení s malým výkonem a používá stejnojmenný programovací jazyk. Java ME prostředí obsahuje množinu standardních Java API, virtuální stroj Javy, uživatelská rozhraní a řadu komunikačních protokolů a zabezpečení. Jelikož jsou mobilní zařízení značně různorodá, tak je programové prostředí daného zařízení definováno profily rozšiřující základní konfigurace. V dnešní době existují dvě základní konfigurace Java ME a to CLDC (Connected Limited Device Configuration) a CDC (Connected Device Configuration). Tyto konfigurace zahrnují minimalizovanou množinu tříd, knihoven a virtuální stroj. Profil je několik API, které jsou definovány na specifické vlastnosti konkrétního zařízení a jeho uživatelské prostředí. Kupříkladu pro mobilní telefony se kombinuje CLDC s profilem MIDP (Mobile Information Device Profile). I při vývoji v Java ME se lze připojovat k databázím, používat webové služby a využívat bezdrátových přenosů. V tomto

případě se musí Java ME rozšířit o dané balíčky, které budou potřebné při vývoji aplikace.

### ***3.3 Vývojová prostředí a programování pro WM***

#### **3.3.1 Visual Studio 2008**

Visual Studio (VS) je komplexní vývojový nástroj. Slouží pro programování stolních, webových, serverových a pro nás zajímavých mobilních aplikací, které se budou spouštět na platformě Windows. K dispozici jsou všechny jazyky platformy .NET, či C a C++. Visual studio je od verze 2005 rozděleno do několika produktových řad. Jmenujme tedy VS Standard Edition, VS Professional Edition , VS Team System a odlehčenou verzi Express Edition, která je k dispozici zdarma. Visual Studio 2008 podporuje následující platformy:

- .NET Compact Framework 3.5
- .NET Compact Framework 2.0
- .NET Compact Framework 1.0 (projekty v této verzi budou aktualizovány na verzi 2.0)
- Windows Mobile 6 (po instalaci Windows Mobile 6 SDK Refresh)
- Windows Mobile 5.0
- Pocket PC 2003 (řízený a nativní kód)
- Smartphone 2003 (pouze nativní kód)
- Windows CE 6.0
- Windows CE 5.0

S Visual studiem 2008 přichází i .NET Compact Framework 3.5, Device Emulator Manager verze 3 a SQL Server Compact Edition 3.5. Novinkami v .NET CF 3.5 je podpora LINQ a implementace Windows Communication Foundation, které umožní komunikaci device-device a nebo device-server přes Exchange Activesync transport. Přístup k databázím pomocí jazyka LINQ umožňuje LINQ to Objects, LINQ to XML a LINQ to DataSet. V současné době není dostupné LINQ to Entities

Device Emulator Manager běží nezávisle na VS 2008 a přehledně ukazuje, jaké typy emulací můžeme provádět, a umožní nám doinstalovat další typy emulátorů



Device emulátor nově umožňuje testovat scénáře reálného dění zařízení jako je pokles síly signálu a stav a vybíjení baterie.

VS 2008 má nově podporu SQL Server CE 3.5 a hlavně vlastní designer (navrhuje grafické rozhraní) pro snadnou tvorbu aplikací spolupracujících s SQL CE 3.5.

Ve VS 2008 máme v rámci Device Security Manageru k dispozici správu bezpečnostních nastavení na zařízení. Můžeme tyto nastavení exportovat/importovat ať už se jedná o fyzické zařízení či emulátor.

Device Certificate Manager nám zase umožňuje prohlížení, vkládání a mazání certifikátů na zařízení.

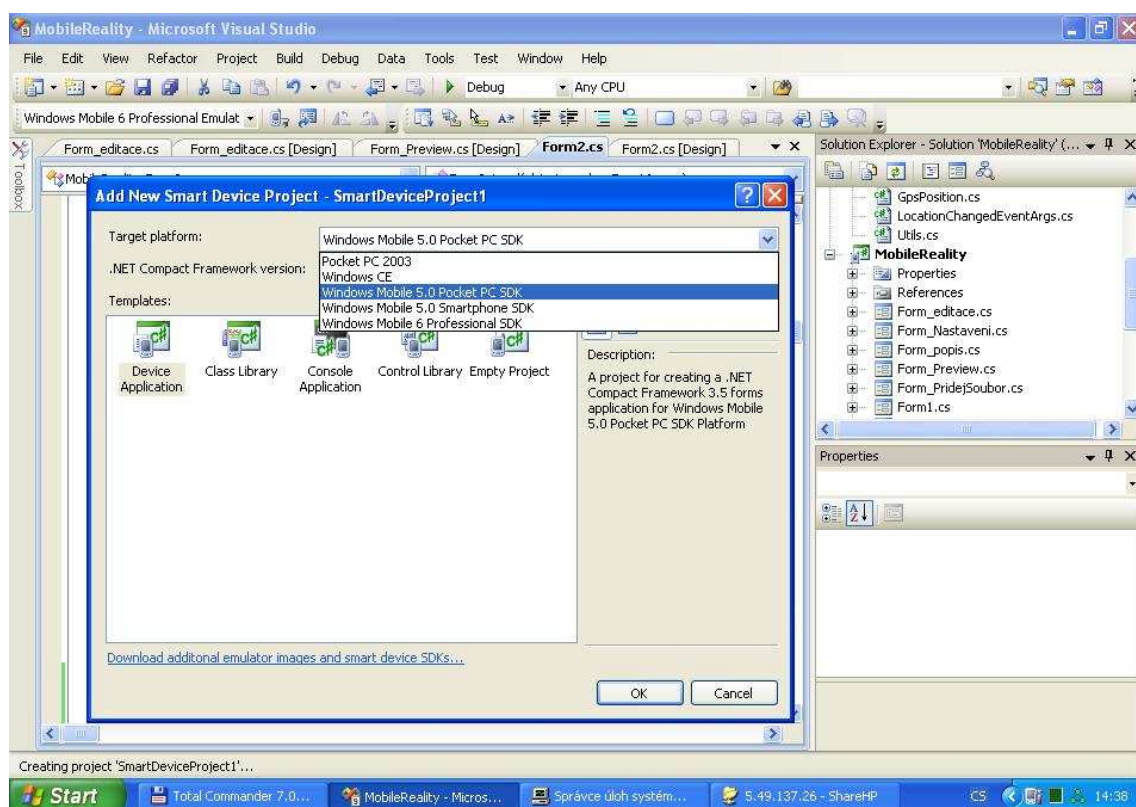
Platforma	Procesory	Varianty
Windows CE 4.2	ARM,MIPSII.,MIPS IV. SH4, X86	
Windows CE 5	ARM,MIPSII.,MIPS IV. SH4, X86	
Windows CE 6	ARM,MIPSII.,MIPS IV. SH4, X86	
Windows Mobile 2003	ARM	Pocket PC Pocket PC Phone Edition
Windows Mobile 5.0	ARM	Pocket PC Pocket PC Phone edition Smartphone
Windows Mobile 6.0	ARM	Classic Standard Professional

**Tabulka 3 Platformy, procesory a varianty WM podporované VS 2008**

### 3.3.2 Vývoj mobilních aplikací ve VS 2008

Názorný příklad vzhledu IDE Visual Studio, vytváření nového projektu a s tím související výběr platformy je vyobrazen na obrázku 6. Nyní bude stručně popsán. Chce-li se začít vytvářet nová aplikace pro WM, stačí ve VS založit nový projekt, vybrat si programovací jazyk, typ *Smart Device* a potvrdit název. Poté, co se potvrdí

jméno, tak se dialogové okno zeptá, zdali se bude vytvářet aplikace s formuláři, konsolová aplikace, *control library*, ale hlavně pro jakou cílovou platformu a cílový .NET Compact Framework se bude programovat. Jestliže nejsou doinstalovaná WM 6 SDK, tak je zde připraven URL odkaz, odkud je možné si případná další SDK stáhnout. Zobrazí se mobilní zařízení a formulář dle typu platformy, která byla zvolena. V *toolboxu* je velké množství ovládacích prvků tak, jak je zvykem z vývoje aplikací ve Windows Forms. Když je aplikaci spuštěna, VS se zeptá, na které zařízení má aplikaci nasadit. Je zde na výběr několik druhů emulátorů a případně i fyzické zařízení. Poté dochází k připojování k emulátoru či fyzickému zařízení a v případě, že je naše aplikace první, tak dochází k instalaci .NET CF.



Obr. 6 : Prostředí a Smart Device projekty VS 2008

### 3.3.3 Windows Mobile 6 SDK

Cílem Windows Mobile SDK je zjednodušení vývoje aplikací a jejich nasazení. S instalací WM SDK se získává především dokumentace, příklady zdrojových kódů, hlavičkové soubory, speciální knihovny, image emulátorů a další nástroje pro Visual

Studio. Vyvinutá aplikace by měla běžet na řadě různých zařízení. A právě WM SDK má zjednodušit její vývoj. Existuje celá řada WM SDK obsahující v sobě již několik emulátorů. Instalace VS 2008 již zahrnuje Windows Mobile SDK 6.0a. Je-li cílem vývoj aplikace pro Windows Mobile 6, je zapotřebí doinstalovat WM SDK6 . WM SDK 6 je dostupné ve dvou verzích a to Standard nebo Professional. Verze Professional ovšem neobsahuje verzi Standard. Pro testování a používání aplikace i na platformě Smartphone je nutné doinstalovat i verzi Standard. Nyní budou stručně shrnuty možnosti obou verzí:

❖ Windows Mobile 6 Standard SDK

- Windows Mobile 6 Standard (176x220 pixelů - 96 dpi)
- Windows Mobile 6 Standard Landscape QVGA (240x320 pixelů - 131 dpi)
- Windows Mobile 6 Standard QVGA (320x240 pixelů - 131 dpi)

❖ Windows Mobile 6 Professional SDK

- Windows Mobile 6 Classic (240x320 pixelů - 96 dpi)
- Windows Mobile 6 Professional (240x320 pixelů - 96 dpi)
- Windows Mobile 6 Professional Square (240x240 pixelů - 96 dpi)
- Windows Mobile 6 Professional Square QVGA (320x320 pixelů - 128 dpi)
- Windows Mobile 6 Professional Square VGA (480x480 pixelů - 192 dpi)
- Windows Mobile 6 Professional VGA (480x640 pixelů - 192 dpi)

Minimální požadavky pro instalaci obou SDK jsou:

- Windows XP SP2, Windows Vista Windows Server 2003
- MS Visual Studio 2005 Standard a vyšší
- MS Compact Framework 2 SP2
- ActiveSync 4.5 a vyšší

### ***3.4 Activesync a Windows Mobile Device Center***

Pro synchronizaci kapesního zařízení s PC a ale i vývoj aplikací je třeba synchronizační software Activesync. Obdobu tohoto programu existuje již od počátků OS pro kapesní počítače společnosti Microsoft. Tento software je standardně dodáván na instalačním CD WM zařízení. ActiveSync umožňuje synchronizaci kontaktů, událostí v kalendáři, textových zpráv, emailů atd., což je výhodné v případě, že bude proveden na WM zařízení Hard Reset(tvrdý reset) a tak se přijde o veškerá data. ActiveSync nám dále umožňuje instalaci a od-instalaci programů na WM zařízení pohodlně skrze PC. Dále můžeme procházet paměť WM zařízení a to nejen interní

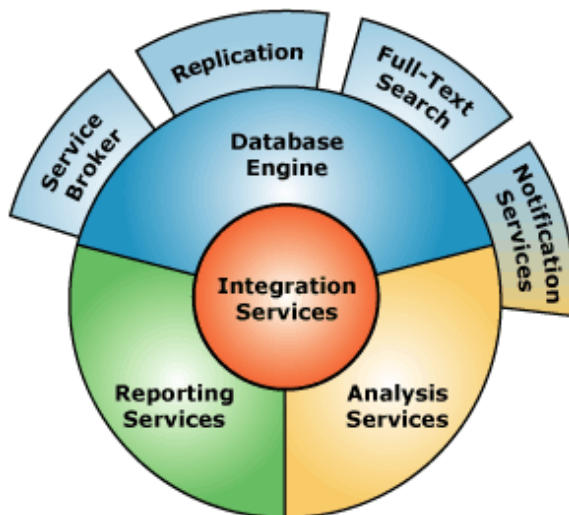
Flash ROM, ale i SD paměťovou kartu. Soubory můžeme kopírovat, přesouvat, přejmenovávat nebo mazat. ActiveSync dále převádí soubory stolního kancelářského balíku MS Office do podoby pro mobilní zařízení Mobile Office a nazpět. Poslední verzí ActiveSync je 4.5, která přidala podporu zařízení s WM 6. S příchodem Windows Vista Microsoft nahradil ActiveSync nástrojem Windows Mobile Device Center, který má tytéž schopnosti, ale lze jej spustit pouze na OS Vista. Naopak ActiveSync je použitelný jen pro Windows XP. Microsoft již nebude dále vyvíjet ActiveSync a zaměří se pouze na Mobile Device Center.

### 3.5 MS SQL server 2008

MS SQL server primárně poskytuje službu databázového serveru. Existuje v několika edicích a to Enterprise, Standard, Workgroup, Web, Developer a volně přístupné verze Express a Compact. Plná verze Enterprise MS SQL poskytuje i několik dalších základních služeb jako:

- Analytické služby
- Notifikační služby
- Reportovací služby
- Služby to transformaci údajů
- Vývojářské a administrátorské nástroje, další komponenty a dokumentace

Na následujícím diagramu Obr. číslo jsou vidět vztahy mezi jednotlivými komponentami SQL Serveru.



## **Obr. 7 : Architektura MS SQL**

SQL Server Workgroup Edition obsahuje základní databázové funkce, ale nezahrnuje další služby. Verze Web edition je zaměřena na Web hosting. Developer edice je totožná s edicí Enterprise, nesmí být použita pro komerční účely, tedy jen pro vývoj a testování například studenty. Standard verze se od plné verze liší hlavně podporou menšího množství instancí. Pro projekty týkající se mobilních aplikací je zajímavá verze Compact.

Po instalaci MS SQL serveru jsou k dispozici také vývojářské nástroje SQL Server Management Studio a Business Intelligence Development Studio. Management Studio je integrovaný a komplexní nástroj pro práci s databázovými objekty a správu MS SQL serveru. Lze pomocí něj přehledně graficky vytvářet a spravovat databáze, tabulky, uživatelská oprávnění, publikace, vložené procedury, trigery atd.. Lze jej použít samozřejmě jako nástroj pro ladění SQL příkazů. Business Intelligence Development Studio je Visual Studio s dalšími typy projektů, které jsou typické pro SQL Server.

### ***3.6 MS SQL Server CE***

MS SQL Server Compact Edition(SQL CE) je kompaktní relační databáze společnosti Microsoft primárně určená pro aplikace běžící na mobilních zařízeních. Je ji možné používat i na stolních počítačích. Dříve byla představována jako SQL Server for Windows CE či SQL Server Mobile Edition. Nejnovější verze SQL CE 3.5 podporuje platformu . NET Framework 3.5 a operační systémy Windows Mobile 2003, 5.0 a 6.0. Tato verze též nativně podporuje 32-bit a 64-bit platformy. SQL CE sdílí společné API s ostatními edicemi SQL Serveru. Také obsahuje ADO .NET služby pro přístup k datům pomocí ADO NET API, synchronizační schopnosti stejně jako LINQ a Entity Framework. Podobně jako velké SQL servery podporuje SQL CE transakce, referenční integritní omezení, zabezpečení a vícenásobné připojení k databázi. SQL CE je limitováno 256 vícenásobnými připojeními. V současné verzi nejsou podporovány vložené procedury a nativní XML datový typ. Dotazy jsou zpracovávány pomocí optimalizovaného dotazovacího procesoru. SQL CE podporuje ještě indexování, vzdálenou replikaci dat (lokální „kešování“ ze vzdálené databáze) a Merge replikace (obustraná synchronizace s velkým MS SQL serverem). Mezi novinky poslední verze

SQL CE patří podpora nových datových typů v SQL Server 2008, jako date, time, datetime2, datetimeoffset, geography a geometri.

SQL CE může být vytvářena a administrována buď pomocí Visual Studia nebo z SQL server Management Studia. SQL CE databáze je koncipována jako jediný SDF soubor, který může být veliký až 4 GB a je možné ho šifrovat 128 bitovým šifrováním. Jedním z hlavních rozdílů SQL CE oproti velkým MS SQL je, že SQL CE není spuštěna jako služba. Runtime a knihovny SQL CE jsou velké přibližně 1.4 MB. Výhodou SQL CE jsou především tedy její velice nízké nároky na paměť a místo.

Bylo by vhodné upozornit na fakt, že SQL Management Studio 2005 může přistupovat pouze k SQL CE databázím 3.0 a 3.1 a verze 2008 téhož programu umí číst SQL CE databáze verze 3.5. V případě Visual Studia 2008 je možné spravovat taktéž SQL CE databáze 3.5 a novější. SQL CE 3.5 sp1 je součástí instalace MS SQL 2008 a Visual Studia 2008 sp1. SQL CE je také volně dostupná na stránkách společnosti Microsoft.

### ***3.7 Mobilní aplikace a databáze***

Databázové aplikace jako takové potřebují pro svůj chod neustálé připojení s databázovým serverem. Při vytváření mobilních aplikací s využitím databáze je možné pracovat ve dvou základních režimech. První režim bude nazván Online, neboť je při něm zapotřebí stálého připojení k databázi. V případě offline režimu se jedná o občasné připojení do sítě. Připojení nemusí být totiž vždy dostupné. Ať už se jedná např. o výpadky ve spojení. Jinou možnost představují pracovníci, kteří se musí pohybovat v terénu a nejsou tedy vždy připojeni, nebo máme databázové aplikace na mobilních zařízeních či na mobilních PC.

Pokud bychom chtěli vytvářet databáze, které nejsou neustále spojené, ale běží v občasném připojení nebo v offline režimu, vyžadují lokální databázový systém, který bude fungovat i jako výkonná cash (vyrovnávací paměť). Aplikace vlastně pracuje s lokálním databázovým systémem a poté díky nějakým synchronizačním službám dojde po připojení našeho zařízení do sítě k synchronizaci dat s velkou firemní databází. Pokud se právě bavíme o těchto scénářích tak máme k dispozici mobilní databázi SQL CE 3.5 a co se týče synchronizační služby, společnost Microsoft nabízí služby Microsoft Synchronization Services.

### ***3.8 Komunikace mobilní aplikace s MS SQL***

Co se týká datového přístupu, lze použít OLEDB Provider, nebo je k dispozici SQL CE DataProvider pro ADO.NET, který obsahuje SQL CE DataReader, DataSety, SQL CE ResultSet podporuje LINQ atd. Je-li řeč o SQL CE ResultSetu, jedná se o objekt, který je nejméně náročný, hlavně co se týká konzumace paměti. Tento objekt má přímý přístup k datům, nemá žádné buferování (ukládání do mezipaměti), ukládání informací při této komunikaci a je možné dokonce udělat aktualizovatelný ResultSet, který provádí přímou aktualizaci dat, což ale nebývá dost zvykem. Většinou aktualizace probíhá tak, že se provede aktualizace lokálně, poté spuštěním nějaké operace ta aktualizace probíhá na serveru. SQL CE ResultSet je schopen dělat aktualizace přímo na velkém SQL Serveru. [12]

### ***3.9 Nasazení aplikací pro SQL CE***

Nejjednodušší cestou, jak nasadit aplikace vytvořené pro SQL CE, je využít nasazení metody click-once. Máme dvě možnosti, buď nasadíme pouze databázi, tzn. datový soubor. Zde je podmínka, že SQL CE musí být už nainstalován na cílovém zařízení a SQL CE může být aktualizován naprosto separátně, běží separátním způsobem. Nebo je možné nasadit databázi včetně knihoven pro běh SQL CE serveru. V tomto případě jsou knihovny součástí aplikace, tzn. aplikace se nainstaluje a sní i knihovny. Pokud se aplikace odinstaluje, odinstalují se i knihovny. Poté různé aktualizace, opečovávání a práce o server je na nás. Aplikace se musí také postarat o běh serveru. Co se týče databází, tak ty se mohou vytvářet až při prvním startu aplikace. Není třeba, aby se databáze instalovala, pokud si člověk při prvním startu aplikace skriptem nebo programovým kódem tuto databázi vytvoří.

### ***3.10 Možnosti synchronizace MS SQL CE s MS SQL***

Pokud pracujeme s lokálním nebo malým mobilním serverem, je možné synchronizovat data tohoto malého serveru s velkým serverem. Jsou tři možnosti. Merge replikace, Remote Data Acces a Sync services for ADO .NET.

Merge replikace s velkým SQL Serverem, je silný nástroj, kde je ale potřeba databázového administrátora, který dobře nastaví pravidla a bude spravovat synchronizace. Tady existuje ale omezení v podobě možnosti replikace pouze s MS SQL serverem. S žádným jiným databázovým serverem toto nelze.

Druhá možnost je Remote Data Access, což je přístup opačný, naprosto programátorský, který využívá přístup přes IIS Server, vše si vlastně vývojář dělá sám. Zde jsou omezené možnosti pro škálovatelnost toho řešení, omezené možnosti pro distribuované řešení více aplikací.

Sync services for ADO .NET je synchronizační služba, která spadá pro Microsoft Synchronization Framework, kdy vývojář využívá jen runtime pro synchronizaci a využívá provider pro ADO .NET, který už zajišťuje synchronizaci dat mezi malým SQL Serverem a velkým SQL Serverem. Toto řešení je jako takové připraveno pro škálovatelnost a hlavně je připraveno i pro SOA Architecture.[12]

### ***3.11 Vzdálený přístup k databázi pomocí ADO .NET***

ADO.NET je rozsáhlá knihovna tříd, která umožňuje práci s databázemi a datovými soubory. Jelikož se jedná o součást .NET Framework, nabízí ADO.NET stejné prostředky, jaké obsahuje samotný .NET (podpora různých jazyků, automatické uvolňování paměti, OOP). Předpokladem pro realizaci ADO.NET je, že databáze není na stejném zařízení jako aplikace, tudíž je spojení s databází přes ADO.NET navázáno pouze po dobu zpracování dotazu. Výhoda spočívá v možnosti připojení více uživatelů v relativně krátkém časovém okamžiku, nevýhodou je udržení konzistence dat v paměti a v databázovém systému. Středem pozornosti v ADO .NET je tzv. „odpojený model“. Tento model funguje následovně. Nejprve se vyšle dotaz na databázi, pomocí něj se naplní objekt *DataSet* a s tím se poté pracuje jako s normální databází. Po provedení změn v *DataSet* se provede zpětná synchronizace.

#### **3.11.1 Zprostředkovatelé přístupu k databázi**

Microsoft při návrhu ADO.NET předpokládal použití s MS SQL Serverem. Programátor může při této kombinaci využívat speciální datové typy. Jedním z těchto typů je zprostředkovatel pro MS SQL SqlClient (System.Data.SqlClient), pro ostatní databáze se jedná především o OleDb. SqlClient je kompletně napsán v řízeném kódu a



při připojení k databázi se snaží využívat co nejmenší počet vrstev. Důsledkem by měl být fakt, že ve většině případů je *SqlClient* rychlejší než *OleDb*. Výhoda *OleDb* spočívá v tom, že to jsou vlastně třídy a metody těchto tříd pro práci s objekty, kde jsou uložena data. Nejedná se přitom pouze o databázi, nýbrž i o soubory či o XML struktury. Další variantou je možnost připojení pomocí ODBC ovladače, který je již však značně zastaralý a nevýkonný.

### 3.11.2 Architektura ADO.NET

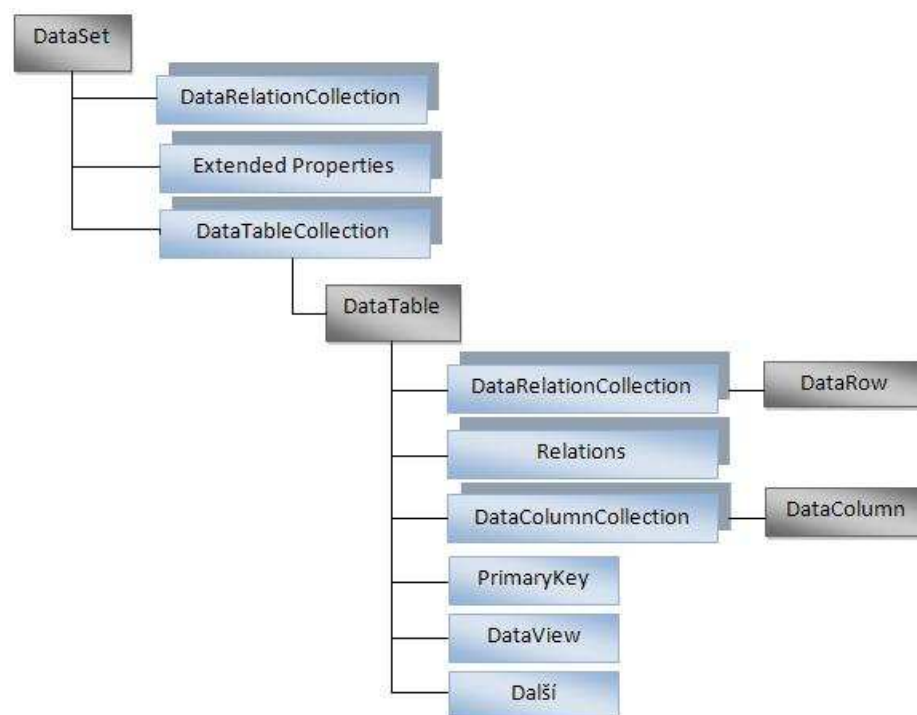
K dispozici je několik základních jmenných prostorů, které jsou potřebné pro práci s databázemi. Za zmínku stojí jistě tyto dva následující prostory:

- *System.Data.OleDb* – třídy zprostředkovatele *OleDb* (pro práci s jinými servery než MS SQL Server)
- *System.Data.SqlClient* – třídy zprostředkovatele MS SQL Server

Použije-li se *OleDb*, prokáže se hlavní přínos v tom, že se jedná o obecný jmenný prostor obsahující třídy pro práci s jakýmkoliv z databázových serverů (takže i MS SQL) nebo datových úložišť. Není-li jisté, jak budou data uložena, předpokládá se využití tříd tohoto jmenného prostoru. Třídy v knihovně ADO.NET je možno rozdělit na dvě množiny.

První množinu tvoří třídy, které jsou zaměřené na správu dat (Data Related Components). Tyto jsou zcela nezávislé na použité databázi. Zároveň tyto třídy reprezentují umístění dat z databáze v paměti.

Z obrázku číslo 8 jasně vyplývá stavební kámen objektu uloženého v paměti. Jedná se o *DataSet*, který reprezentuje databázi ve formátu XML a jako s databází lze s tímto objektem pracovat. Tím je odstraněn problém nepřístupnosti dat. Je možné programově přistoupit k jakékoliv tabulce uložené v *DataSetu*, dále potom definovat primární klíč tabulky či vztahy mezi jednotlivými tabulkami.



**Obr. 8 : Struktura DB dat v paměti**

Druhá množina je tvořena třídami, které jsou závislé na použité databázi (Data Provider Components). Tyto třídy umožňují propojení mezi třídami pro správu dat a databázovými systémy. Mezi nejdůležitější se řadí:

- *SqlConnection* – navázání spojení s databází
- *SqlCommand* – provádění SQL dotazů
- *SqlDataReader* – reprezentuje nejjednodušší přístup k datům v databázi. Lze zmínit určité nevýhody –např. pracuje pouze s připojenou databází a v jednom okamžiku umožňuje práci pouze s jedním řádkem v databázi. Výhodou je, že pro práci nepotřebuje DataSet.
- *SqlDataAdapter* – datový adapter, slouží pro přenos dat z databáze do DataSetu

Lze říci, že ke každé z těchto tříd je možno nalézt ekvivalent pro práci s jinou databází než Microsoft SQL.



**Obr. 9 : Komunikace Mobilní aplikace MS SQL**

Na základě rozhraní ADO.NET lze připojit k programu libovolný databázový systém, který disponuje vlastní *OleDb* ovladačem. Přehled systémů, které se dají běžně použít je:

- MS SQL Server
- Oracle – konkurenční server k MS SQL Server
- MySQL – open – source určený pro multiplatformní použití
- Microsoft Access – určený pro tvorbu jednouživatelských databází

Je možné také použít XML. XML je využíváno při většině interaktivních operací. Data v XML lze načítat do DataSetu či naopak data z DataSetu je možné ukládat v XML formátu. Na základě schémat dokumentu XML je program schopen automaticky vygenerovat zdrojový kód v příslušném programovacím jazyce.

### ***3.12 Merge synchronizace***

Merge replikace se používá především ve chvíli kdy má více uživatelů pořizovat data v offline režimu. Popišme si zásady a princip tvorby mobilní aplikace, která bude využívat, lokální databáze na PDA a merge replikace pro synchronizaci s velkým SQL serverem. V zásadě tento postup můžeme popsat několika kroky:

1. Konfigurace publikací SQL Serveru
2. Konfigurace IIS(Internet Information Services) pro replikace
3. Vytvoření SQL CE Subscription
4. Tvorba mobilní aplikace.

### **3.12.1 Konfigurace Publikací**

Nejprve je nutné vytvořit buď pomocí designeru (návrhového nástroje) a nebo pomocí skriptu samotnou databázi dle našich potřeb. V této chvíli se musí připravit server pro publikaci dat. Nejprve je třeba vytvořit uživatelský účet a sdílenou složku, do které budou nahrávány snapshot soubory. Tato složka musí mít nastavená příslušná sdílení a práva čtení a zápisu pro daný uživatelský účet, pomocí kterého k ní budeme přistupovat. Nyní je možné přistoupit k samotnému vytvoření publikace. Využijte se k tomu SQL Server Management Studio. V uzlu replikace se klikne pravým tlačítkem na Local Publications a vytvoří se pomocí průvodce novou publikací. V průvodci se postupně volí daná použitá databáze, vyberou se Merge publikace a tabulky, které budou publikovány.

Také je třeba nastavit přístupová práva k dané databázi. To lze provést opět pomocí Management Studia, kde v Object Exploreru, v položce Security se klikne pravým tlačítkem na Logins a vybere se New Login. Poté se spustí dialog box, kde se vybere pro autentifikaci uživatelský účet, který byl vytvořen za tímto účelem a nastaví se mu role db\_owner. Musí se také nastavit stejným způsobem přístup pro IIS anonymního uživatele a přidat ho do Publications Access List (PAL).

### **3.12.2 Konfigurace SQL a IIS Web Serveru pro webovou synchronizaci**

Samotné konfiguraci publikací předchází instalace SQL Serveru, pokud již není nainstalován. Konfigurace publikací pro webovou synchronizaci je prováděna opět pomocí Management Studia. V uzlu Publikace se vybere vytvořená publikace a pravým tlačítkem se klikne na Configure Web Synchronization. Tím se opět spustí průvodce, v němž se nastavuje Subscriber Type a Web Server, což budou SQL CE a počítač, na kterém běží IIS. Jako Alias ve Virtual Directory Information obrazovce je zapotřebí vybrat databázi. Ve Snapshot Share Acces obrazovce se vybere sdílená složka. Pro ověření funkčnosti nastavení se vloží do Internet Exploreru následující adresu: <http://localhost/SQLMobile/sqlcesa30.dll?diag>

### **3.12.3 Konfigurace SQL CE**

Dříve než se začne vytvářet samotnou aplikaci, musíme nastavit SQL CE. Opět je využíváno Management Studio k vytvoření nové databáze. Tentokrát se vytváří

databázi pro mobilní zařízení. Novou subscription a publication je možné vytvořit stejným způsobem, jako v kapitole 3.12.1, tedy pomocí průvodce.

Nyní je vše připraveno a nakonfigurováno a můžeme se pustit do vytváření samotné aplikace.

### ***3.13 Programování webových aplikací***

Abychom zde uvedli alespoň stručný přehled technologií, které slouží k tvorbě dynamických webových stránek, zmíníme zde technologie PHP a ASP.

#### **3.13.1 PHP**

PHP vzniklo v roce 1996 a znamená Personal Home Page. Od té doby tento pojem prodělal mnoho změn a nyní je PHP chápáno jako Hypertext Preprocessor. PHP představuje programovací jazyk, který pracuje na straně serveru (kde jsou uloženy zdrojové kódy webových stránek). Je určen především pro programování dynamických webových stránek. Nejčastěji se objevuje začlenění přímo do struktury jazyka HTML, XHTML či WML, což lze následně využít při tvorbě webových aplikací. Programovací jazyk byl inspirován několika programovacími jazyky (Perl, C, Pascal a Java) a stal se tak oblíbeným jazykem především díky své jednoduchosti použití a tomu, že kombinuje vlastnosti těchto výše jmenovaných programovacích jazyků.

PHP je možné použít i k tvorbě konzolových a desktopových aplikací. S PHP lze měnit, mazat a ukládat data webových stránek. PHP skript je nejdříve proveden na serveru a poté je prohlížeči odeslán pouze výsledek. JavaScript se naopak vykonává přímo v prohlížeči. Dalším rozdílem je, že zdrojový kód PHP na rozdíl od JavaScriptu a HTML nelze zobrazit. Na základě PHP je možné vytvořit diskuzní fórum, knihu návštěv, počítadlo, anketu, graf. PHP zároveň umožňuje propojení webových stránek s databázemi, např. MySQL, ODBC, Oracle, PostgreSQL, MSSQL), podporu celé řady internetových protokolů (HTTP, SMTP, SNMP, FTP, IMAP, POP3, LDAP...)

Webová stránka s prvky PHP má nejčastěji koncovku .php. Lze však použít i .php3, php4, php5 a phtml. PHP je jazyk, který si nevystačí jen s prohlížečem určité verze (na rozdíl od HTML či JavaScriptu). PHP vyžaduje instalaci na počítači. Základem je webový server a knihovny. K podpoře PHP se doporučuje instalace a konfigurace serveru Apache. Pro snadnou instalaci je nejvhodnějším řešením pro

instalaci PHP využití programu PHP Triad či easyPHP, který vše sám nainstaluje. Je třeba však upozornit na fakt, že ne každý webhosting zahrnuje podporu PHP. Tato podpora je nadstandardní službou, která je za příplatek. Při důkladném hledání je však možné nalézt i webhosting zdarma i s podporou PHP.

## **Výhody PHP**

Výhody PHP možné charakterizovat v následujících bodech

- PHP jazyk je specializovaný na webové stránky
- rozsáhlý soubor funkcí v základní knihovně PHP + dalších z PECL
- nativní podpora mnoha databázových systémů
- další nativně implementované funkční celky snadno instalovatelné z repozitáře PEAR
- možnost provozu na více OS (zejména Linux, Windows)
- další knihovny instalovatelné z repozitáře PECL
- možnost využití nativních funkcí operačního systému (možná nekompatibilita s jiným OS)
- zpočátku strmá křivka učení
- obrovské množství projektů a kódů, které lze zdarma využít (WordPress, phpBB a další)
- dobrý, aktuální a úplný manuál s příklady – postačující dokumentace
- velmi svobodná licence – svobodnější, než GPL
- stručný přístup k prvkům přes hash-mapy

## **Nevýhody PHP**

- jazyk PHP není nikde definován, je popsán pouze jeho implementací
- mírně nekonzistentní vývoj v minulosti, který si sebou PHP nese dosud (někdy často měnící se příkazy atd...)
- nekonzistentní pojmenování funkcí a nejednotné pořadí parametrů
- ač jazyk výborně podporuje výjimky, jeho knihovna je používá jen zřídka
- slabší podpora Unicode, pouze přes PHP knihovnu (v PHP 6 má být Unicode řetězec jako základní typ)
- neumožňuje překlad do byte kódu, PHP skript se při každém požadavku překládá znovu
- dosud neexistuje formální specifikace syntaxe jazyka a jeho chování (jedinou úplnou definicí je interpreter)
- ve standardní distribuci chybí ladící (debugovací) nástroj
- po zpracování požadavku neudrží kontext aplikace, vytváří jej vždy znovu (oslabuje výkon)
- nepodporuje paralelní výpočet a synchronizaci (vlákna)

### 3.13.2 ASP .NET

ASP (Active Server Pages) je další technologií vyvinutou společností Microsoft. Jedná se o skriptovací prostředí pro servery, které lze využít pro tvorbu a provozování dynamických, interaktivních webových aplikací. S využitím *ASP*, je možné kombinovat *HTML* stránky, skripty, a *COM* komponenty a vytvářet tak interaktivní webové stránky nebo výkonné webové aplikace, které se snadno vyvíjejí i upravují. Microsoft uvedl na trh novější verzi ASP, tzv. ASP .NET. ASP .NET tedy není jen další verzí ASP, ve skutečnosti se jedná o zcela novou technologii. Je to sada softwarových technologií společnosti Microsoft pro propojování světa informací, lidí, systémů a zařízení. Tato nová verze poskytuje zcela nový model programování pro vytváření síťových aplikací používajících výhod sítě Internet. Základem je již zmiňovaná platforma *.NET Framework*. Na rozdíl od staré technologie ASP, ve které se programovalo strukturálně, je ASP .NET typický svým objektově orientovaným programováním. Základními stavebními kameny ASP .NET jsou: *Web Forms*, *Web Services* a *Mobile Internet Toolkit*. *Web Forms* slouží k vytváření webových stránek pomocí komponent podobně, jako je tomu zvykem u „okénkových“ aplikací. *Mobile Internet Toolkit* je sada nástrojů sloužících ke generování výstupu pro mobilní zařízení.

### 3.13.3 Porovnání PHP a ASP

Přestože se jedná o dvě konkurenční technologie od různých výrobců, nabízejí v podstatě obdobné možnosti. První rozdíl je patrný v tom, že *ASP* je spíše objektově orientované a *PHP* je realizováno velkým množstvím samostatných funkcí. *PHP* je subtilní samostatná technologie, která vývojářům poskytuje poměrně dobrou infrastrukturu, sama se ale o nic většího neopírá. Naproti tomu je *ASP.NET* součástí rozsáhlého *.NET Frameworku* a samostatně nemůže existovat. Speciálně vytvořené *PHP* je přímočarou platformou, která bez okolků poskytuje účelové nástroje pro vývoj webových aplikací. Nejen proto je *PHP* často označováno jako „jednoduché“. *ASP.NET* je naopak součástí víceúčelové platformy. Rozsáhlost *.NET Frameworku* tvoří občas problémy začátečníkům – *ASP.NET* působí na první dojem složitě (v kontrastu s jednoduchostí *PHP*). Ačkoliv tedy každý přístup má svá pro a proti, výhodou *PHP* je, že se ho lze kvůli jeho jednoduchosti rychle naučit.

## 4. Analýza problému

Jedním z cílů této diplomové práce bylo navrhnout způsob záznamu snímků pomocí mobilního zařízení a vytvořit aplikaci zpracovávající dané snímky. Při hledání uplatnění daného problému v praxi bylo možné se ubírat mnoha směry. Praktických příkladů aplikací v praxi, které by mohli využívat mobilního zařízení, která bude tyto snímky vytvářet a dále s nimi manipulovat, je celá řada.

Pro názornost je v této práci uveden příklad realitní kanceláře, kde by daného navrženého systému využívali především makléři při své realitní činnosti. Používáním navrženého systému firma ušetří nejen čas, nýbrž může dojít i k úspoře finančních prostředků. Následuje popis fungování RK. Na základě seznámení se s fungováním realitní kanceláře se čtenář dozví mnoho informací, které mu pomohou pochopit význam a fungování navrhovaného systému.

### 4.1 Běžný chod realitní kanceláře

Každá realitní kancelář má svůj systém. Vesměs jsou ale principy všech kanceláří podobné a záleží spíše na jejich zaměření a portfoliu působení. Pro konkrétní příklad chodu realitní kanceláře můžeme například použít *Realitní Centrum Candia*.

*Realitní Centrum Candia* se zaměřuje především na prodej pozemků a bytů. Má však ve své nabídce i pronájmy jak bytů, tak nebytových prostor a kanceláří. Činnost realitní kanceláře se dá popsat jak na běžném pracovním týdnu, tak na jednom obchodním případě. Na základě prvního kontaktu sekretariátu se zákazníkem se dohodne makléř s majitelem na setkání, na které nesmí přijít pozdě. S sebou by si měl vzít firemní desky, firemní propisku, nabídku realitní kanceláře, ukázkou dosavadní inzerce, aby seznámil klienta s postupem, co se bude dít. Klientovi je třeba zdůraznit, že v případě jakýchkoliv dotazů se kdykoliv může obrátit na kontaktní osobu. Makléř sděluje obecný postup, jak probíhá naběr nemovitosti a další postup co se právního servisu týče. Cílem makléře je nafotit si danou nemovitost a zjistit o ní veškeré potřebné údaje, které se nejen píšou pro zákazníky na internet a do různých forem inzerce, ale tvoří i důležitý poklad pro získávání dalších dokumentů potřebných pro vypracování případných smluv či podkladů nutných k získávání doplňujících informací od jiných institucí.



Tyto informace, které se zaznamenávají do náběrového listu poté makléř předává sekretářce, která zakládá složku k dané nemovitosti a tvoří inzerát, který je umístěn na vlastní webové stránky a další realitní servery, se kterými realitní kancelář spolupracuje. Důležité je vždy vytvořit první formu inzerátu, která se nenásledně kopíruje. Jak bylo uvedeno, sekretářka si zakládá složku dané nemovitosti, ve které nesmí chybět text inzerátu, základní identifikační a kontaktní údaje na majitele nabízené nemovitosti, fotky a veškeré dokumenty, týkající se dané nemovitosti (např. výpis z katastru nemovitostí, různé druhy smluv a jiné spisy). Tuto složku poté předá makléřovi, který si ji přiřadí do svého portfolia ke svým stávajícím zakázkám.

## ***4.2 Princip systému***

Nyní je vidět, jak funguje obecně realitní kancelář. Je možné tedy vysvětlit princip celého systému a jeho zapojení do fungování RK. Systém MobileReality, jak byl pojmenován autorem, by měl fungovat následujícím způsobem. Každý makléř RK bude mít k dispozici vlastní mobilní kapesní zařízení MDA. To již v praxi funguje, neboť makléři MDA používají k poznámkám a využívají jej jako GPS navigaci. Poté, co sekretariát předá kontaktní informace klienta makléřovi, dohodne si makléř schůzku se zákazníkem.

Na schůzce s klientem již nastupuje systém MobileReality. Makléř místo vyplňování papírového náběrového listu spustí na svém MDA aplikaci MobileReality a vyplní elektronickou obdobu náběrového listu. Dále vyplní kontaktní a identifikační údaje o majiteli. Jeho úkolem je také dle potřeb nafotit daný objekt. K tomu opět slouží aplikace MobileReality, jelikož je z ní možné přistupovat k fotoaparátu MDA. V případě, že má makléř ve svém MDA přístupné další nafocené dokumenty, může je také pomocí MobileReality přiřadit k záznamu o klientovi.

Důležitou novinkou, kterou s sebou MobileReality přináší, je možnost přesné lokalizace dané nemovitosti pomocí GPS modulu, s nímž umí MobileReality spolupracovat a jednoduše jej ovládat. V některých případech není vhodné GPS souřadnice zveřejňovat, což samozřejmě systém podporuje. V průběhu náběru nemovitosti může makléř i po vyplnění informací tyto údaje upravovat. Náběr nemovitosti končí potvrzením a odesláním veškerých informací a dat na SQL server. Aplikace automaticky využije dostupné připojení k internetu. Přenosy dat mohou

probíhat pomocí bezdrátových technologií Wifi, GPRS,EDGE či UMTS nebo pomocí kabelového ActiveSync využívajícího spojení MDA s notebookem připojeným do internetu.

Nesmírně důležitou výhodou je však makléřův neustálý přehled o veškerých nemovitostech. Ve svém kapesním počítači má skrze MobileReality přístupnou databázi nemovitostí, rozdělenou na základě kritérií, která jsou v realitní kanceláři používána a může tak kdykoliv podat komukoliv aktuální informace i o jiných nemovitostech. To je důležité především při prohlídkách nemovitostí, kdy zájemce není spokojen například s nabízenou prohlíženou nemovitostí a měl by zájem o další informace i o jiných nemovitostech. Makléř tak může snadno a rychle podat kompletní přehled o jiných nabízených nemovitostech.

Součástí systému je i webová aplikace vytvářející inzeráty na webové prezentaci kanceláře. Jakmile jsou veškerá data odeslána na databázový SQL server, vygenerují se také automaticky s tím inzeráty na webových stránkách realitní kanceláře.

Na základě aplikace a užívání navrhnutého komplexního systému by mělo dojít k urychlení práce, snížení nákladů a efektivnímu využívání MDA v realitní kanceláři. Výhodou je i úspora času sekretářky, která nemusí zadávat inzerát sama na internet, stačí pouze umístění inzerátu na jiné realitní servery.



**Obr. 10 : Princip systému MobileReality**

### ***4.3 Analýza požadavků na systém***

Při vývoji systému a hlavně mobilní aplikace bylo nutné vést konzultace s lidmi, kteří by daný systém využívali. Bylo nezbytné zjistit informace o tom, jaké funkce navrhovaného systému by realitní kancelář využila, jak by měla být koncipována aplikace pro nejefektivnější a intuitivně ovladatelné grafické rozhraní. Důležité bylo zjistit, jaká data se budou na SQL server ukládat a posléze pouze vybraná data přenášet na webovou prezentaci RK. Myšleny jsou tím například citlivé informace o majiteli a v případě některých nemovitostí i jejich lokalita a další údaje, které kvůli konkurenčnímu boji RK nezveřejňují. Při návrhu a vývoji systému je potřeba konzultovat řešení s potenciálními uživateli. Existuje celá řada jednotlivých specifik realitních kanceláří. Stejně by tomu bylo i u mobilní aplikace.

### ***4.4 Nutné technické požadavky na systém***

Jelikož je systém MobileReality komplexní řešení využívající vícero hardwarových a softwarových komponent, vznikají tím i určité požadavky na použitý hardware a omezení, týkající se použité softwarové platformy jak pro běh aplikace, tak i pro její vývoj. V následujících kapitolách jsou tyto požadavky a omezení detailněji popsány.

#### **4.4.1 Mobilní zařízení**

Aplikace MobileReality používá několik hardwarových komponent, které by mělo dané kapesní zařízení obsahovat, či by měla existovat možnost rozšířit zařízení o tyto komponenty. Je-li řeč o zařízení jako takovém je třeba pro běh aplikace zařízení typu PDA/MDA. Daná aplikace by měla také jít spustit na Smartphone, ale její primární používání je navrženo pro dotykový displej.

#### **GPS modul**

Pro zjišťování polohy dané lokality je třeba, aby mobilní zařízení disponovalo GPS modulem. GPS moduly využívající čip SIRFstar III a vyšší jsou již velice přesné. Je nutné, aby operační systém nativně podporoval toto zařízení a byly dostupné ovladače pro daný systém. Neměl by být problém s žádným GPS modulem

kompatibilním s operačním systémem Windows Mobile díky tomu, že aplikace využívá wrapper (knihovna zabalující funkce pro použití v C#) využívající GPS API pro Windows Mobile. Taktéž lze díky této vlastnosti používat GPS modul i jinou aplikací.

## **Kamera**

Pro pořizování snímků objektů je nutné, aby kapesní počítač měl k dispozici interní či externí kameru. V případě interní kamery se předpokládá, že budou existovat ovladače pro daný OS a kamera bude spolupracovat se standardním Camera API pro Windows Mobile. Většina zařízení obsahuje i software pro danou Built-in kameru, který rovněž využívá Camera API pro WM. V některých případech se může stát, že se výrobce zařízení záměrně snaží, aby nebylo možné s interní kamerou spolupracovat a tak například prostřednictvím klíčů v registru či jiným způsobem toto znepřístupní či omezí funkce Built-in kamery ovládané skrze API. Tento fakt byl zjištěn při vývoji pro konkrétní zařízení Eten Glofiish M750 a řešení tohoto problému na daném zařízení bylo věnováno dosti času s téměř nulovým výsledkem. Při testování aplikace na emulátoru a dalších zařízeních byl běh aplikace bezproblémový. Posléze byly zkoumány i možnosti nápravy pro konkrétní zařízení Glofiish M750.

Aplikace MobileReality má schopnost používat i externí kamery. Zde není možné využít Camera API pro WM. Hardwarové požadavky na externí kameru byly již popsány v kapitole 1.1.4. Co se týče samotného mobilního zařízení, je nutné, aby mělo USB host periférii k připojení externí kamery. Pro ovládání kamery jsou nutné ovladače pro daný hardware. Zde nastává zásadní problém v jejich nedostupnosti. Jediným řešením je, v případě použití externí UVC kamery připojené přes USB host, použití vlastních ovladačů. MobileReality využívá za tímto účelem knihovnu MobileCamera, která je založená na principu projektu Windows CE Webcam. Tím pádem jsou podporovány pouze vybrané USB kamery. V případě použití dalších USB kamer je nutno přidat do daného ovladače podporu výše zmíněných kamer.

Proto, aby mohla aplikace odesílat data je nutné připojení k internetu, realizované libovolným způsobem. To může být zprostředkováno buď pomocí Wifi modulu, GSM/GPRS/EDGE modulu či UMTS modulu využívajícího datových přenosů třetí generace. Vzhledem k přenosu multimediálních dat autor doporučuje použít technologii s minimální přenosovou rychlostí odpovídající standardu EDGE.

Pro pohodlnější vyplňování formulářů aplikace je výhodné, aby MDA zařízení disponovalo hardwarovou klávesnicí.

#### **4.4.2 OS mobilního zařízení a používaný software**

Systém MobileReality je určený pro mobilní zařízení s operačním systémem Windows mobile 5.0 a vyšší. Aplikaci lze spustit i na nižších verzích OS, ale není zaručena funkčnost spolupráce s GPS. V reálném provozu byla testována na Windows Mobile verze 6 a 6.1. Testy probíhaly také na emulátorech starších verzí systému i na nadcházející verzi 6.5.

Pro běh aplikace MobileReality je nutné mít nainstalován na mobilním zařízení Compact Framework 3.5. Vlastní aplikace by si vystačila i s verzí 2, ale v případě používání SQL CE databáze přímo na zařízení a její synchronizace s MS SQL 2008 je třeba mít nainstalovánu verzi 3.5.

Jelikož v aplikaci není prozatím používána možnost použití lokální databáze v kapesním počítači, není ani používána synchronizace v podobě merge replikace a s tím související nutnost instalace SQL CE na MDA a PC reprezentující SQL server.

Jako úložiště dat získaných prostřednictvím mobilní aplikace byl vybrán databázový server Microsoft SQL Server 2008. Mohl být použit i jiný databázový systém, ale ten by nemohl využívat všech výhod platformy .NET. Navíc v případě použití merge synchronizace je podporován pouze MS SQL Server. Používaný databázový systém je nutno provozovat na PC s operačními systémy Windows XP Professional SP2, Windows Server 2003, Windows Vista, či Windows Server 2008. Pro komunikaci SQL CE 3.5 a MS SQL Server 2008 není podporován Windows XP Home edition z důvodu absence IIS (Internet Information Services).

Předpokládá se, že SQL server má veřejnou adresu IP, aby k němu bylo možné přistupovat prostřednictvím sítě internet. V případě, že není k dispozici veřejná IP adresa, je možné použít volně dostupné VPN (Virtual Private Network) klienty. Příkladem může být produkt Hamachi, který je dostupný jak pro desktop PC, tak i pro kapesní zařízení s Windows Mobile.

## ***4.5 Výběr platformy a vývojových prostředků***

Při výběru mobilního zařízení a s tím souvisejícího operačního systému bylo jednoznačně vybráno kapesní zařízení typu MDA s operačním systémem Windows Mobile. MDA zařízení založená na Windows Mobile jsou svými vlastnostmi předurčená spíše k business účelům. Systém Windows Mobile je v tomto segmentu velmi rozšířen a oproti ostatním systémům má tedy mnohem větší podporu pro vývoj aplikací směřujícím k pracovním účelům.

Jako vývojový nástroj bylo zvoleno Visual Studio 2008, které se jeví jako nejvhodnější volba při psaní mobilních aplikací v .NET CF. VS 2008 s sebou přináší kromě technologie IntelliSense (pro efektivnější psaní kódu), podporu několika emulátorů a řadu nástrojů, usnadňujících vývoj pro mobilní zařízení. Programovací jazyk byl vybrán C#, který je v mnohém podobný jazyku Java.

Při psaní aplikace byl využíván vývojový balíček Windows Mobile 6 Professional SDK a Windows Mobile 6 Standard SDK. Oba balíčky přinášejí kromě ukázkových zdrojových kódů také emulátory WM 6. Před instalací SDK bylo nutné nainstalovat též program ActiveSync 4.5, který slouží pro komunikaci s reálným MDA zařízením.

## ***4.6 Nezbytné konfigurace***

### **4.6.1 Konfigurace SQL Serveru**

Jedním z kroků, které je třeba provést před nasazením systému MobileReality do provozu je instalace a nastavení MS SQL Serveru 2008. Při instalaci je vhodné nastavit uživatelská oprávnění a potřebné služby SQL serveru jako instance SQL Serveru, SQL Server Browser a SQL Server Agent. Pro pozdější změnu nastavení služeb lze použít SQL Server Configuration Manager. Také je nutné povolit TCP/IP protokol pro spuštěnou instanci serveru, což lze provést opět pomocí Configuration Manageru.

Je třeba nastavit i bránu firewall. Následuje návod, který je popsán v jednotlivých krocích, jako příklad nastavení brány firewall systému Win XP. Výchozí instance Serveru používá port 1433, je nutné v okně nastavení brány firewall přidat TCP port 1433 a povolit výjimku pro službu Sqlservr.exe.

Cesta k danému souboru základní instance serveru MS SQL Server 2008 je:  
C:\Program Files\Microsoft SQL Server\MSSQL10.MSSQLSERVER\MSSQL\Binn\Sqlservr.exe

Po správné konfiguraci Sql Serveru je třeba vytvořit a nastavit databázi s tabulkami.

### **Vytvoření databáze**

Databáze se vytvoří za pomoci SQL Management Studia níže popsaným způsobem. Po spuštění SQL Management Studia a následném přihlášení k databázovému systému účtem systémového administrátora, se ve stromové struktuře *object exploreru* pravým tlačítkem klikne na položku *tables* a vybere se *new table*. V nově otevřeném okně se nastaví jméno databáze, což bude *mobilereality*. Dále je v témže formuláři důležité nastavit seznam vlastníků databáze a jejich práva.

### **Vytvoření tabulek databáze**

Tabulky lze vytvářet za pomoci *designeru*, což je výhodné především u tvorby první tabulky. *Designer* se spustí přes místní nabídku položky *Tables* databáze *mobilereality*. Dle potřeby se vytvoří požadovaný počet sloupců a datové typy.

Vytvořené tabulky je možno zálohovat v podobě skriptu, který lze jednoduše nechat vygenerovat. Další tabulky je výhodnější vytvářet pomocí úpravy vygenerovaných skriptů.

Databáze *mobilereality* obsahuje v testovacím provozu prozatím 5 tabulek. I přesto, že byly prováděny analýzy potřeb realitní kanceláře, na základě kterých byly vytvořeny příslušné tabulky, je nutné říci, že autor předpokládá pozdější změnu ve struktuře databáze. Pro základní funkce celého systému MobileReality používaný návrh plně postačuje. Struktura obsahuje tabulky *souhrn*, *foto*, *makleri*, *nemovitost a oblast*. V tabulce *souhrn* jsou ukládány souhrnné informace o klientovi a transakci. Tabulka *foto* uchovává fotografie pořízené prostřednictvím MDA zařízení. Informace o makléřovi, který prováděl naběr nemovitosti, jsou obsaženy v tabulce *makleri*. Poslední dvě tabulky jsou navrženy tak, aby z nich bylo možné získat data související s typem nemovitosti, a oblastí ve které se nachází nemovitost (realitní kancelář působí ve více oblastech).

### Příklad skriptu zálohy tabulky foto:

```
CREATE TABLE [dbo].[foto](
    [ID_FOTO] [int] IDENTITY(1,1) NOT NULL,
    [ID_SOUHRN] [int] NOT NULL,
    [FOTO] [image] NOT NULL,
    [FILE_SIZE] [int] NOT NULL,
    CONSTRAINT [PK_foto] PRIMARY KEY CLUSTERED
(
    [ID_FOTO] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

GO

ALTER TABLE [dbo].[foto] WITH CHECK ADD CONSTRAINT [FK_foto]
FOREIGN KEY([ID_SOUHRN])
REFERENCES [dbo].[souhrn] ([ID])
GO

ALTER TABLE [dbo].[foto] CHECK CONSTRAINT [FK_foto]
GO
```

Tabulka *foto* slouží pro ukládání multimediálních dat. Prozatím jsou ukládány pouze obrázky. K jedné transakci tabulky *souhrn* může existovat více obrázků. Proto je v tabulce *foto* cizí klíč *ID\_SOUHRN* odkazující na příslušný záznam v tabulce *souhrn*. Do sloupce *FOTO* jsou ukládány data typu *image*, reprezentující samotnou fotografii. Poslední sloupec *FILE\_SIZE* je datového typu *int* a slouží k uchovávání informací o velikosti souboru. Hodnoty uložené v tomto sloupečku jsou využívány při načítání fotografie z databáze na web. *ID\_FOTO* je nastaveno na automatickou inkrementaci a nemusí být proto zapisováno z mobilní aplikace. Všechny hodnoty v tabulce nesmí být prázdné tedy *NOT\_NULL*.

Bylo by nevhodné zde uvádět skripty vytvářející ostatní tabulky databáze. Tabulky byly vytvářeny a jsou zálohovány podobným způsobem, jako zmiňovaná tabulka *foto*. Tento názorný příklad pouze ilustruje, jak byla tvořena celá databáze.

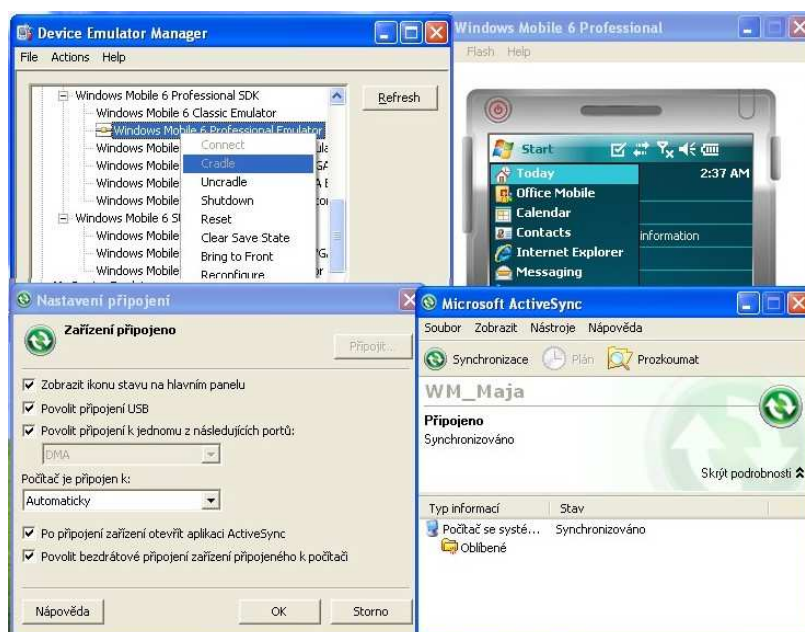


## Konfigurace SQL pro použití s lokální databází v MDA

Aplikace MobileReality byla testována i s možností ukládání dat do místní databáze v MDA. V tomto případě je v MDA lokální sdf soubor (databáze) a ten je merge replikací synchronizován s databází velkého SQL Serveru. Pro komunikaci SQL CE v MDA s SQL Serverem je používána IIS (Internetová informační služba). Tu je nutno společně s SQL serverem nakonfigurovat. Detailně je postup konfigurace rozebrán v kapitole 2.12, věnované merge replikaci.

### 4.6.2 Konfigurace emulátoru

Při vývoji aplikace pro mobilní zařízení byly také používány emulátory. Ty je nutno správně nakonfigurovat, aby mohly prostřednictvím sítě internet komunikovat s databázovým serverem. Propojení bude realizováno za pomoci ActiveSync. V aplikaci ActiveSync je třeba v nastavení připojení povolit připojení pomocí DMA. Tím je zajištěno, že se ActiveSync bude chovat vůči emulátoru jako by to bylo skutečné zařízení. To, aby se choval emulátor jako reálné zařízení vůči ActiveSync se musí nakonfigurovat v Device Emulator Manageru. V Device Emulator Manageru se spustí požadovaný emulátor a pomocí příslušné místní nabídky se nastaví jako Cradle (připojený do kolébky).



Obr. 11 : Konfigurace ActyveSync a emulátoru

## 5. Vývoj aplikace MobileReality

### 5.1 Popis tříd a knihoven

*Form1.cs*- úvodní obrazovka

*Form2.cs*- ústřední formulář aplikace, provádí se na něm většina práce s aplikací, zpřístupňuje kameru, GPS, náběrový list a přikládání dalších souborů v PDA.

*Form\_editace.cs*- zpřístupňuje online náhled do databáze

*Form\_Nastaveni.cs*- provádí nastavení veškerých vlastností kamery, parametrů připojení a informací o makléři.

*Form\_Preview.cs*- slouží k výběru typu nemovitosti (mění hodnotu globální proměnné *TypNemovitosti*). Při vytváření nového záznamu do databáze, spouští hlavní pracovní formulář *Form2*. Je-li použit ve scénáři náhledu do databáze, spustí formulář *Form\_editace* zprostředkovávající data SQL serveru.

*FormPopisBytProd.cs*

*FormPopisBytPron.cs*

*FormPopisNebytProd.cs*

*FormPopisNebytPron.cs*

*FormPopisPozemek.cs*

*FormPopisRD.cs*

Formuláře náběrového listu, slouží pro zadávání informací o klientovi a vyplnění potřebných informací o nemovitosti.

*Globalni\_Nastaveni.cs*- třída určující základní nastavení aplikace, obsahuje struktury a vnořené třídy s globálními proměnnými.

*OvladaniGPS.cs*- třída obsluhující GPS, obsahuje několik metod starajících se o spuštění GPS, vypnutí GPS a načtení GPS souřadnic (*StopGPS*, *StartGPS*, *GetPosition*). Pro předávání stavu GPS je definován delegát *PositionRetrieved*.

*Program.cs*- obsahuje metodu *main*, definuje a inicializuje základní objekty.

*RWRegister.cs*- třída pro práci s registry WM. Obsahuje metody pro zápis a čtení hodnot registru.

*Microsoft.WindowsMobile.Location.dll*- knihovna (Wrappper) tvořící most mezi C# aplikací a nativním GPS API.

*MobileCamera.dll*- ovladač externí kamery

## 5.2 Funkce a scénáře mobilní aplikace

Po spuštění aplikace se zobrazí úvodní formulář *Form1*. Formulář není maximalizovaný, aby jej bylo snadné kdykoli minimalizovat a pokračovat v jiné práci. Uživatel má na výběr čtyři možnosti. Může rovnou začít vytvářet nový záznam, prohlížet záznamy v databázi, měnit nastavení aplikace či program ukončit. *Form1* obsahuje 4 *PictureBox* komponenty, které mají funkci tlačítka a při události *Click* přemístí uživatele na požadovanou funkci. Všechny formuláře, nevyjímaje *Form1*, mají kontextová menu, ve kterých jsou umístěny převážně navigační a potvrzovací funkce. V některých případech obsahují kontextová menu položky, které duplicitně reprezentují funkce ovládacích prvků na formuláři. Menu jsou výhodná především v případě ovládání programu pomocí hardwarových tlačítek PDA/MDA.



Obr. 12 : Úvodní obrazovka

### 5.2.1 Nový záznam

Po stisku tlačítka *Nový* je uživatel přesměrován na obrazovku, která ho vyzývá k výběru typu nemovitosti, kterou chce vkládat. Na výběr má z šesti typů nemovitostí, které je možno zvolit opět pomocí *Pictureboxů* reprezentujících grafická tlačítka.

Nemovitost lze vybrat také kontextovým menu. Menu umožňuje i návrat uživatele na úvodní obrazovku. Po výběru nemovitosti je ověřován přístup k databázovému serveru a v případě úspěšného připojení je uživatel přesměrován na centrální formulář *Form2*. V opačném případě je uživatel upozorněn *MessageBox* zprávou na nemožnost připojení k databázi a při potvrzení je přesměrován na úvodní obrazovku aplikace.

Formulář *Form2* zpřístupňuje uživateli funkce kamery, spuštění GPS, přiložení dalších dokumentů a vyplnění náběrového listu. Formulář ještě obsahuje další prvky jako *Info panel* a *id klienta*. V informačním panelu jsou zobrazovány veškeré provedené akce a čas kdy se uskutečnily.



Obr. 13 : Výběr nemovitosti



Obr. 14 : Ústřední formulář

Grafická tlačítka jsou tvořena čtyřmi *PictureBox* komponentami. Několik *labelů* představuje popisky. Text *labelu* informujícího o stavu GPS modulu je ovládán dynamicky podle stavu GPS. Kontextové menu má navigační funkci zpět a potvrzovací funkci vložit. Data v informačním panelu jsou aktualizována funkcí *UpdateInfoPanel*.

Po stisku tlačítka fotoaparátu (událost *foto\_pictureBox\_Click*) je spuštěn s náležitými parametry *CameraCaptureDialog* pro pořízení fotografie. Vzhled *CameraCaptureDialog* je dán výrobcem MDA a používaným operačním systémem.

GPS modul je aktivován událostí *lokace\_pictureBox\_Click*, která patří tlačítku znázorňujícímu GPS. Řízení GPS zajišťuje knihovna *OvladaniGPS.cs*. a předávání informací o stavu GPS je zajišťováno pomocí události *OnPositionRetrieved*.

K fotografiím, nafoceným přímo z aplikace MobileReality, lze přiřadit i další snímky, které jsou již uloženy v kapesním zařízení z dřívější doby. Snímky lze vybrat stiskem tlačítka vzhledově připomínajícího dokumenty (ofocené výpisy z katastru nemovitosti, stavební povolení aj.). Jedná se o tlačítko *dokumenty\_pictureBox* a událost *dokumenty\_pictureBox\_Click*, která nastavuje a otevírá *OpenFileDialog*. Filtr souborů *OpenFileDialog* je nastaven na soubory typu *jpeg*.



Obr. 15 : CaptureDialog WM 6(emulátor)



Obr 16 : OpenFileDialog WM 6(emulátor)

Vyplnění náběrového listu je možné po stisku příslušného tlačítka znázorňujícího náběrový list. Konkrétně se jedná o tlačítko *popis\_pictureBox* a jeho událost *popis\_pictureBox\_Click*. Metoda *pictureBox\_Click* přesměruje uživatele na formulář náběrového listu příslušného typu nemovitosti. Formuláře všech druhů nemovitostí mají určité společné prvky jako komponenty *tabControl* a *mainMenu*. *MainMenu* v tomto případě opět plní navigační a potvrzovací funkci. *TabControl* zahrnuje ve všech případech dvě záložky *tabPages* (*tabPage1*=Informace o majiteli,

*tabPage2=Popis*). V záložce *Informace o majiteli* se vyplňují kontaktní údaje majitele nemovitosti (Jméno, Příjmení, Telefon, atd.). Položky (*textBoxy*) v záložce *Popis* reprezentují část náběrového listu s možností zadat informace o konkrétní nemovitosti. Počet a význam *textBoxů* se liší dle typu nemovitosti (V případě pozemku jsou typické např. *Výměra, Inženýrské sítě, Orientace, Přístupová cesta*, atd., ...). Při stisku *uložit* (*menuItem2\_Click*) se hodnoty z *textBoxů* uloží do proměnných a v případě znovuotevření náběrového listu jsou již předvyplněny. Metoda *menuItem2\_Click* také kontroluje, zda-li jsou vyplněny důležité vstupní informace, bez kterých by nebylo možné vytvořit inzerát na webových stránkách. Při nevyplnění nezbytných informací je uživatel informován *MessageBoxem* a není mu povoleno pokračovat v další práci.



**Obr. 17 : Náběrový list (popis)**



**Obr. 18 : Náběrový list (Info o majiteli)**

Pro dokončení práce a odeslání veškerých dat na server slouží tlačítko *Vložit*, přesněji metoda *menuItem2\_Click*. Po odeslání dat na server je uživatel obeznámen o úspěšnosti provedené akce.



### 5.2.2 Náhled databáze

Podstatnou výhodou aplikace MobileReality je okamžitý náhled do databáze informací o nemovitostech uložených na SQL Serveru. Procházení databáze je druhým hlavním scénářem aplikace a je přístupné z úvodní obrazovky prostřednictvím tlačítka *Otevřít* (vykonavatelem je metoda *pictureBox2\_Click*).

Otevřít (*PictureBox2\_Click*) přesměruje uživatele na obrazovku s výzvou k výběru nemovitosti. K tomuto účelu je využíván totožný formulář (*Form\_Preview*) jako v případě výběru nemovitosti nového záznamu. Po výběru nemovitosti je kontrolován přístup do databáze a v případě úspěšného připojení je uživateli dostupný formulář *Form\_editace*, sloužící k podrobnému náhledu do databáze. K zobrazování dat je používána komponenta *dataGrid*. Metoda *dataGrid1\_CurrentCellChanged* zajišťuje otevření editačního formuláře příslušejícímu typu prohlížené nemovitosti. Navigace a potvrzování veškerých změn je dostupné skrze *mainMenu*.



The image shows a screenshot of a mobile application interface. At the top, there is a table with four columns: 'Typ', 'Parcelní', 'Výměra', and 'Inženýrské'. The table contains seven rows of data. Below the table, there is a large empty rectangular area. At the bottom of the screen, there is a navigation bar with two buttons: 'Zpět/Zrušit' and 'Uložit'.

Typ	Parcelní	Výměra	Inženýrské
Pozemek	111	654	voda, elektrin
Pozemek	123	2500	voda, elektrin
Pozemek	321	1300	kanalizace, pl
Pozemek	456	1700	voda, elektrin
Pozemek	742	3000	kanalizace, pl
Pozemek	364	990	kanalizace, pl
Pozemek	687	1700	voda, elektrin

Obr. 19 : Online procházení databáze

### 5.2.3 Nastavení aplikace

Z úvodní obrazovky aplikace je možné přistupovat skrze tlačítko *Nastavení* (metoda *pictureBox3\_Click*) do formuláře *Form\_Nastaveni*. Uvedený formulář je graficky rozdělen pomocí komponenty *TabControl* na několik záložek (*Nastavení kamery-tabPage1*, *Nastavení dat-tabPage2*, *Uživatel-tabPage3*).

Nastavení kamery umožňuje zvolit typ kamery (interní/externí), režim kamery (foto, video), defaultní název fotografií a další. Přepínače volby jsou realizovány komponentou *panel* a dvěma *radioButtony*. Text je vkládán do *textBoxů*.

V Nastavení dat je možné změnit název databázového SQL serveru, jméno používané databáze, přihlašovací údaje uživatele k připojení na server atd.

Doplňující identifikační a kontaktní informace o makléři, jako jméno, příjmení a telefon, se vyplňují v záložce Uživatel (*tabPage3*). Tyto údaje je nutné vyplnit.

Zrušení provedených změn a uložení informací se provádí pomocí *mainMenu*. Ukládání veškerých nastavení provádí metoda *menuItem2\_Click*. Hodnoty jsou ukládány do globálních proměnných, aby k nim mohlo být přistupováno ze všech formulářů. Defaultní nastavení aplikace je načítáno z knihovny *Globalni\_Nastaveni.cs*. V případě použití pouze jmenované knihovny by nastavení bylo resetováno při každém spuštění aplikace. Tento problém řeší ukládání důležitých dat do registrů systému. Zápis a čtení hodnot registru řeší knihovna *RWRegister*.



Obr. 20 : Nastavení kamery



Obr. 21 : Nastavení dat



### 5.3 Spolupráce s kamerou

Pořizovat fotografie umožňuje aplikace MobileReality pomocí stisk tlačítka znázorňujícího fotoaparát umístěného na hlavním pracovním formuláři *Form2*. Metoda *foto\_pictureBox\_click*, volaná při stisku tlačítka nejprve zjišťuje, zdali je vybrána interní nebo externí kamera. Je-li zvolena interní kamera, je ověřován přednastavený mód kamery a následně je volán konstruktor *CameraCaptureDialog*. Poté jsou volány příslušné metody a provedou se veškerá nastavení kamery dle podmínek.

Příklad zdrojového kódu názorně ukazuje nastavení interní kamery do režimu focení (rozlišení 640\*480) s nejvyšší kvalitou fotografie.

```
CameraCaptureDialog cameraCapture = new CameraCaptureDialog();
cameraCapture.Owner = this;
cameraCapture.Mode = CameraCaptureMode.Still;
string pictureFileName = Program.nastavKameru.kameraDefaultPictureFileName;
string pictureFileExtension = Program.nastavKameru.kameraDefaultPictureExtension;
cameraCapture.DefaultFileName = pictureFileName+ cisloFoto + pictureFileExtension;
cameraCapture.Resolution = new Size(640, 480);
cameraCapture.StillQuality = CameraCaptureStillQuality.High;
cisloFoto = cisloFoto + 1;
cameraCapture.ShowDialog();
```

Po zavolání konstruktoru jsou postupně nastaveny proměnné *Mode*, *DefaultFileName*, *Resolution*, *StillQuality* a nakonec je vyvolán dialog metodou *ShowDialog*.

V případě, že metoda *foto\_pictureBox\_click* zjistí, že byla vybrána externí kamera, provede se blok zdrojového kódu obsluhující externí kameru. Externí kamera je ovládána pomocí knihovny *MobileCamera.dll*. Knihovnu je nejprve nutno importovat. K importování se použije atribut *DllImport*. Importované funkce musí být označeny jako externí (klíčové slovo *extern*).

Úryvek zdrojového kódu ukazuje příklad importování knihovny a deklaraci externích funkcí.

```
[DllImport("MobileCamera.dll")]
private static extern bool CaptureStill(string Path);

[DllImport("MobileCamera.dll")]
private static extern bool InitializeGraph(IntPtr hWnd);

protected string ImageStoreLocation { get; set; }

bool _running;

private string _currentFile;
```

Následující blok kódu naznačuje práci s externí kamerou. V aplikaci je vykonáván v metodě *foto\_pictureBox\_click* formuláře Form2 za splnění podmínky určující externí kameru. Pořízení fotografie se provádí voláním funkce *CaptureStill* s parametrem *newFileName*.

```
string newFileName = string.Format("obrazek", DateTime.Now.ToString("yyyyMMdd-
hhmmss"));

try
{
    _currentFile = newFileName;
    CaptureStill(newFileName);
    prirazeniFoto = MessageBox.Show("Přiřadit Foto k aktuálním souřadnicím?",
    "Photo", MessageBoxButtons.YesNo, MessageBoxIcon.Question,
    MessageBoxDefaultButton.Button1) == DialogResult.Yes;
    if (prirazeniFoto == true)
    {
        cesty_obrazek.Add(newFileName);
        prirazeniFoto = false;
    }
}
catch
{
    MessageBox.Show("Není připojena externí kamera");
}
```

## 5.4 Ovládání GPS

Aktivace GPS modulu se vizuálně provádí na ústředním formuláři *Form2* stiskem tlačítka znázorňujícího GPS. Pro ovládání GPS je využívána třída *OvladaniGPS.cs*. Je nutno tedy zavolat konstruktor této třídy při inicializaci třídy *Form2*. Po stisku tlačítka je vykonána metoda *lokace\_pictureBox\_Click*.

Výpis kódu metody *lokace\_pictureBox\_Click*:

```
PositionMetadata mt = new PositionMetadata();  
ovladanigps.StartGPS();  
ovladanigps.OnPositionRetrieved += newPositionRetrieved(gpsController_OnPositionRetrieved);  
ovladanigps.GetPosition(mt);  
UpdateInfoPanel(String.Format("GPS: {0}", "čekám na signál..."));  
labelStatus.Text = "Hledám satelity...";
```

GPS data jsou ukládána do objektu *mt*. GPS je startována voláním metody *StartGPS*. Data z GPS jsou získána voláním metody *GetPosition*.

Podtřídy *GetPosition*, *StopGPS*, *gps\_LocationChanged* a *StartGPS* obsaženy ve třídě *OvladaniGPS* volají metody knihovny *Microsoft.WindowsMobile.Location.dll*.

Pro ilustraci je uvedena ukázka implementace v podobě třídy *StartGPS*.

```
public void StartGPS()  
{  
    if (!gps.Opened)  
    {  
        gps.Open();  
    }  
}
```

## 5.5 Komunikace s SQL Serverem

Aplikace MobileReality se v zásadě připojuje k SQL serveru ve třech případech.

Prvním případem je ověřování dostupnosti SQL Serveru při spouštění nejdůležitějšího pracovního formuláře *Form2*. V případě nedostupnosti připojení je uživatel upozorněn na chybu, není mu povolen přístup do požadovaného formuláře a je vrácen na úvodní obrazovku aplikace.

Druhou situací je ukládání dat na server pomocí tlačítka *uložit* formuláře *Form2*. Dle typu nemovitosti je odeslán na server SqlCommand s příslušným CommandTextem. CommandText je obdobný syntaxi jazyka SQL.

Posledním místem, odkud se aplikace připojuje k SQL Serveru, je formulář *Form\_Editace*. Při načítání formuláře dochází k připojení na server, data jsou nejprve uložena do *Datasetu* a následně zobrazena v grafické komponentě *dataGrid*.

Ve všech případech je komunikace ošetřena bloky *try* a *catch* pro případ nedostupnosti serveru. Ve třídách, které se připojují k databázi, je třeba uvést jmenné prostory *System.Data.SqlClient* a *System.Data.SqlServerCe*.

Následuje zjednodušený případ připojení k databázi a naplnění dataSetu daty.

```
string connStr = @"Server="+ServerIp+";Database="+DbName+"; User Id="+UserId+";
Password="+DbPassword+""; //connection string

string s = "select * from dbo.souhrn";

myConn = new SqlConnection(connStr);

myConn.Open();

SqlCommand com = new SqlCommand(s, myConn);

com.ExecuteNonQuery();

SqlDataAdapter myAdapter = new SqlDataAdapter(s, myConn); //naplnění datasetu

DataSet myDataSet = new DataSet();

myAdapter.Fill(myDataSet);

myAdapter.Fill(myDataSet, "Result");
```

## 6. Webová aplikace

Součástí celého systému MobileReality je i webová aplikace. Aplikace je psána pomocí skriptovacího jazyka PHP. Webová část systému MobileReality má dva účely. Prvním účelem je obecná prezentace realitní kanceláře. Druhým, významnějším cílem, je zprostředkování prezentace automaticky generovaných inzerátů nemovitostí z dat uložených na SQL serveru. Pozornost v diplomové práci bude věnována především samotným php skriptům nikoli HTML kódu vytvářejícímu design a rozložení stránek.

### 6.1 Konfigurace web serveru pro přístup k MS SQL

Před nasazením webu je nutné nainstalovat webový server. MobileReality používá server *Apache*, který je součástí balíčku *EasyPHP3.0*. Jeho výhodou je snadná instalace. *Apache* komunikuje s databázovým serverem MS SQL skrze knihovnu *ntwdblib.dll*, kterou je nutné doinstalovat. Důležité je poupravit konfigurační soubor *php.ini* tak, aby byla povolena a nastavena ke komunikaci zmíněná knihovna *ntwdblib.dll*.

Knihovnu je nutno nakopírovat do adresáře externích knihoven. Konfigurační soubor je možné editovat pomocí poznámkového bloku. K souboru lze přistoupit i přes grafické rozhraní *EasyPHP3.0*, které se nachází v kontextovém menu pod položkou *Configuration*. Celý obsah webové prezentace, včetně php skriptů, je umístěn ve složce *www* v adresářové struktuře balíčku *EasyPHP 3.0*.

### 6.2 Přehled php skriptů

*index.php* - skript hlavní stránky zobrazující novinky

*lokalita.php*, *typy\_nemovitosti.php* - volají se při výběru lokality a nemovitosti obsahují odkazy na typy lokalit a nemovitostí.

*objekt.php* – detail jednotlivých objektů

*objekt\_vyber.php* – stará se o zobrazení pouze určitého typu objektu

*pozemek\_vyber.php* – zobrazí pouze pozemky v určité lokalitě, váže se k *lokalita.php*,

*show.php* - obstarává načítání bin. dat ze serveru a optimalizaci čtení fotografií

## 6.3 Odesílání požadavků na databázi

Ve většině php skriptů jsou odesílány SQL dotazy na databázový server. Následuje ukázka komentovaného zdrojového kódu php skriptu, provádějícího inicializaci připojení k serveru MS SQL.

```
<?php
// připojení k serveru na lokálním pc skrze port 1433 a autentizace
$link = mssql_connect('localhost,1433', 'sa', 'heslo');
if(!$link)
{
    die('Nastala chyba během připojení k MSSQL');
}
// výběr databáze mobilereality
mssql_select_db('mobilereality', $link);
?>
```

Jak odeslat požadavek na MS SQL server ukazuje další část zdrojového kódu.

```
<?php
$sql = "SELECT TOP 8 * FROM dbo.souhrn ORDER BY ID DESC";
$data = mssql_query($sql);
while($row = mssql_fetch_assoc($data))
{
    $sql1 = "SELECT * FROM dbo.foto WHERE ID_SOUHRN = ".$row["ID"];
    $data1 = mssql_query($sql1);
    $row1 = mssql_fetch_assoc($data1);
    $img_id = $row1["ID_FOTO"];
}
?>
// na tomto místě je delší HTML kód
<?php
}
?>
```

## ***6.4 Optimalizace načítání fotografií z databáze***

Fotografie nemovitostí jsou ukládány do databáze MS SQL Serveru. V případě, že by byla fotografie při každém zobrazení stránky znovu přenášena z databáze, byl by SQL server velice zatížen. Navíc data nemohou být načtena najednou, nýbrž po menších blocích, které je posléze nutno spojit. Proto byla vytvořena optimalizace načítání dat, která zajišťuje, že jsou všechny nové fotografie po prvním načtení zkopírovány na ftp a při dalším zobrazení jsou již čteny z ftp. Načítání dat a optimalizaci zajišťuje skript *show.php*.

## 7. Závěr

Cílem této diplomové práce byl návrh způsobu záznamu obrazu pomocí interní/externí kamery mobilního zařízení. Dále měla být vytvořena aplikace manipulující s uloženými snímky. Nejprve bylo ovšem nutné se seznámit s problematikou mobilních zařízení, vybrat vhodnou platformu a typ kapesního počítače, který by umožňoval záznam snímků a jejich následnou manipulaci. Po zvážení a prostudování všech východisek, bylo zvoleno kapesní zařízení typu MDA (Mobile Digital Assistant), disponující všemi potřebnými prostředky a technologiemi jako interní kamera, GPS, GPRS/EDGE, Wifi. Studií byly také zjištěny požadavky na kapesní počítač a externí kameru v případě, že by zařízení nedisponovalo kamerou interní. Jako nejpříhodnější platforma z hlediska podpory a zaměření se jeví Windows Mobile.

Pro dané téma bylo nejvhodnější si zvolit praktický příklad využití. Za tímto účelem byl vytvořen komplexní systém sloužící potřebám realitní kanceláře. Hlavní přednosti navrhovaného řešení vyplývají ze samotného nápadu nasazení systému v podmínkách RK a daly by se popsat následujícím způsobem. Systém by měl být užíván již od fáze získávání potenciálního zákazníka, přes možnost neustálého přístupu do databáze stávajících nemovitostí, které může makléř nabízet svým klientům až po samotnou prezentaci inzerátů na firemních webových stránkách. Díky využití modulu GPS je možné lokalizovat nemovitost a odeslat její souřadnice spolu s údaji o nemovitosti a majiteli přímo z prostředí aplikace skrze všudypřítomné GPRS/EDGE připojení do databázového serveru. Od počátku uvedení této aplikace do firemních procesů tak dochází k úspoře času i financí dané realitní kanceláře.

Obecný princip navrženého řešení by našel uplatnění i v řadě dalších aplikací. Příkladem mohou být profesní dokumentace stavebních projektů či automaticky aktualizovaná webová prezentace cestovatelských zážitků.

Případných vylepšení systému by mohlo být také mnoho. Možné rozšíření projektu autor vidí ve spolupráci systému s google maps.



## Seznam použité literatury

- [1] Lavička, O. *SDIO - Připojte ke svému PDA skener či čtečku prstů* [online]. [2009-05-05]  
URL : [http://palmare.idnes.cz/sdio-pripojte-ke-svemu-pda-skener-ci-ctecku-otisku-prstu-pk7-/pda.asp?c=A050131\\_5293763\\_pda](http://palmare.idnes.cz/sdio-pripojte-ke-svemu-pda-skener-ci-ctecku-otisku-prstu-pk7-/pda.asp?c=A050131_5293763_pda)
- [2] Wikipedia. *Windows CE* [online]. [2009-05-05]  
URL: [http://en.wikipedia.org/wiki/Timeline\\_of\\_Microsoft\\_Windows](http://en.wikipedia.org/wiki/Timeline_of_Microsoft_Windows)
- [3] Microsoft Corporation. *Windows Embedded CE* [online]. [2009-05-05]  
URL: <http://msdn.microsoft.com/en-us/library/ms905093.aspx>
- [4] Šejda, J. *J2EE, .NET a vývoj rozsáhlých systémů 2* [online]. [2009-05-10]  
URL : <http://interval.cz/clanky/j2ee-net-a-vyvoj-rozsahlych-systemu-2/>
- [5] Troelsen, A. *C# a .NET 2.0 profesionálně*, Zoner Press, Brno 2006
- [6] Horváth, T. *.NET Framework* [online]. [2009-10-05]  
URL : <http://programujte.com/index.php?akce=clanek&cl=2008120700--net-framework>
- [7] Mahmoud, H. *Naučte se Java 2 Micro edition*, Grada, Praha 2006
- [8] MUNI Wikipedia. H. : *Java ME*, [online]. [2009-05-12]  
URL: [http://kore.fi.muni.cz:5080/wiki/index.php/Java\\_ME](http://kore.fi.muni.cz:5080/wiki/index.php/Java_ME)
- [9] Microsoft Corp. *What's New for Develop. in WM6* [online]. [2009-05-12]  
URL: <http://msdn.microsoft.com/en-us/library/bb278115.aspx>
- [10] WindowsForDevices. *VS 2008 adds mobile app. features* [online]. [2009-05-13]  
URL: <http://www.windowsfordevices.com/news/NS2201413774.html>
- [11] Price, J. *C# programování databází*, Grada, Praha 2005
- [12] Microsoft Corp. *MSDN Webcasty* [online]. [2009-05-17]  
URL: <http://www.windowsfordevices.com/news/NS2201413774.html>
- [13] Egea, M. *Microsoft SQL Server 2005: základy databází: krok za krokem*, Computer Press, Brno 2007